ESD-TR-76-366

DETAIL SPECIFICATION FOR THE SECURITY PROTECTION MODULE (SPM)

Honeywell Incorporated
Aerospace Division
13350 U.S. Highway 19
St. Petersburg, FL 33733

September 1976

Prepared for

DEPUTY FOR COMMAND AND MANAGEMENT SYSTEMS
ELECTRONIC SYSTEMS DIVISION
HANSCOM AIR FORCE BASE, MA 01731

LEGAL NOTICE

When U.S. Government drawings, specifications or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

OTHER NOTICES

Do not return this copy. Retain or destroy.

"This technical report has been reviewed and is approved for publication."

William R. Price

WILLIAM R. PRICE, Captain, USAF
Project Engineer/Scientist

Donald P. Eriksen

DONALD P. ERIKSEN
Project Engineer/Scientist

FOR THE COMMANDER

Frank J. Emma

FRANK J. EMMA, Colonel, USAF
Director, Computer Systems Engineering
Deputy for Command & Management Systems

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-76-366	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DETAIL SPECIFICATION FOR THE SECURITY PROTECTION MODULE (SPM)		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER 477-14498
7. AUTHOR(s) G. Rolfe J. Carnall		8. CONTRACT OR GRANT NUMBER(s) F19628-74-C-0193
9. PERFORMING ORGANIZATION NAME AND ADDRESS Honeywell Incorporated Aerospace Division 13350 U.S. Highway 19, St. Petersburg, FL 33733		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Deputy for Command and Management Systems Electronic Systems Division Hanscom Air Force Base, MA 01731		12. REPORT DATE September 1976 13. NUMBER OF PAGES 95
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public Release; Distribution Unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
This specification defines the technical requirements for a Security Protection Module (SPM). An SPM is the hardware portion of a security kernel whose function is to mediate, through a descriptor structure, all interactions between elements of a protected minicomputer. The SPM evaluates the propriety of all requests, and performs address translation between virtual requests and physical resources. The SPM ccntains a fast-access cache for storing copies of descriptors in an effort to minimize the performance overhead associated with security.

PREFACE

Because of funding limitations, the Air Force terminated the effort which this document describes before the effort reached its logical conclusion. This specification has not been formally approved, but was published in the interest of capturing and disseminating the computer security technology that was available when the effort was terminated.

This specification was prepared in accordance with MIL-STD-490. The reader unfamiliar with the Automatic Data Processing Security Program (sponsored by the Air Force Electronics Systems Division) may find the format difficult to read and for more background want to refer to the "Analysis of Secure Communications Processor Architecture" and "Secure Communications Processor Specification" (ESD-TR-76-351, Vol. I and II, respectively).

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF FIGURES

1.0 SCOPE

1.1 General

This specification defines the performance, design, development, and test requirements for the Security Protection Module (SPM). The SPM shall provide the hardware required to convert a Honeywell Level 6 minicomputer into a certifiably secure communications processor. The requirements contained within this document are Level 6 specific and are derived from the generic requirements contained in the Secure Communications Processor Specification prepared under contract F19628-74-C-0205.

Other system element requirements including the security software (security kernel), are beyond the scope of this document, but are defined in the SFEP Subsystem Specification.

2.0 APPLICABLE DOCUMENTS

2.1 General Applicability

The following documents form a part of this specification to the extent specified herein. In the event of a conflict between the documents specified herein and the content of this specification, the content of this specification shall be considered a superseding requirement.

2.2 Military Specifications and Standards

MIL-STD-130		Identification Markings of U.S. Military Property
MIL-STD-461A	1 Aug. 68	Electromagnetic Interference

2.2 Military Specifications and Standards (Continued)

MIL-STD-1000		Drawings, Engineering and Associated Lists
MIL-STD-1472A	15 May 70	Human Engr. Design Criteria for Military Systems, Equipment and Facilities
MIL-STD-454	Rev. D 31 Aug. 73	Standard General Requirements Electronic Equipment
MIL-STD-756	Rev. A 15 May 63	Reliability Prediction
MIL-HDBK-217B		Reliability, Stress, and Failure Rate Data for Electronic Equipment
MIL-E-5400	Class 1	General specification for Aircraft Electronic Equipment
MIL-S-901C		Shock Test, High Impact, Shipboard Machinery, Equipment and System Requirements for (NAVY)
NACSEM 5100	Oct. 70	Compromising Emanations Laboratory Test Standard Electromagnetic
AF&C DH 1-4		Electromagnetic Compatibility

2.3 Honeywell Documents

60126298	Rev. C, 1 Jan. 75	Engineering Product Specification for Minicomputer Bus
		• Aero Design Procedure
		• BCO Standard Parts List

2.3 Honeywell Documents (Continued)

60130050	Rev. B 18 Jun. 76	Engineering Product Specification for NML Inter System Link
TBS		SCOMP Product Assurance Plan

2.4 Other Documents

Contract F19628-74-C-0205	• Secure Communications Processor Architecture Study
Contract F19628-74-C-0205	• Secure Communications Processor Specification • SFEP Subsystem Specification

3.0 REQUIREMENTS

The equipment specified herein shall be designed in accordance with the requirements of this specification.

3.1 SPM Definition

3.1.1 SPM Functional Overview

The function of an SPM is to mediate, through a descriptor structure, all interactions between elements of a protected minicomputer. The logical structure that the introduction of an SPM imposes on the protected minicomputer is diagrammed in Figure 1. An SPM is intimately associated (for purposes of SPM control) with each processor of the system. Through its SPM, each processor may communicate with the other processors, I/O devices and memory. An I/O device may communicate to memory through an SPM and returns status to the processor that initiated its current operation. Thus, each SPM may be thought of as an address translation

3.1.1 SPM Functional Overview (Continued)

resource for a number of requestors, the requestors being the attached processors and I/O devices. The address translation operation is the conversion of virtual addresses presented by the requestors, via the descriptor structure, to absolute resource addresses (using information contained in the descriptors).

Each SPM logically contains the mechanism diagrammed in Figure 2. It contains the following items:

1. The current protection state (current and effective ring) of each requestor it services.
2. A pointer (Descriptor Base Root) to the set of descriptors which describe the accessible resources for each requestor.
3. A mechanism by which the protection state and set of resource descriptors may be initialized for each requestor: this mechanism is generally under the control of the associated processor.
4. A mechanism by which the SPM may search through the descriptor structure to locate the proper descriptor applying to a requested resource.
5. A mechanism by which the SPM may evaluate the propriety of a requested access based on the following information: the identity of the requestor, the access mode of the request, the resource requested, the current protection state of the SPM for the requestor, and the requestor's descriptor for the resource.

3.1.1 SPM Functional Overview (Continued)

6. A mechanism by which the protection state of a requestor may be changed, in a well-defined manner.
7. An internal cache in which the SPM may place fast access copies of recently referenced descriptors.

The SPM shall mediate each request by a processor to:

1. Reference memory
2. Initiate an I/O operation

The SPM shall be capable of mediating each request by an I/O device to reference memory.

The SPM's active mediation of I/O requests may cause an unacceptable performance loss (particularly in high I/O bandwidth applications). Thus an alternative form of I/O mediation is specified. This architecture, termed pre-mapped I/O, imposes substantially more responsibility and complexity on the I/O controller certification. Thus its use in secure systems must be carefully considered. Pre-mapped I/O mediation imposes a "one-time" check of the propriety of the I/O devices memory requests. This checking, equivalent to the dynamic checking discussed above, is performed at I/O initiation time. The virtual memory address and extent to or from which the I/O device is to transfer data, is transmitted as data to the SPM which interprets the addresses in the descriptor structure of the requesting processor. These addresses, if valid, are then translated into absolute addresses and transmitted

3.1.1 SPM Functional Overview (Continued)

to the device. The device must be guaranteed not to modify the addresses passed to it. The SPM must guarantee that the processor does not modify the set of descriptors used in the translation until the completion of the I/O operation.

Physically, the SPM consists of two major components. One component is the Virtual Memory Interface Unit (VMIU) that is physically mounted on the CPU in the slot normally used by the Memory Protection Unit. The VMIU is functionally between the CPU address register and the bus and will mediate all CPU direct memory requests.

The remaining portion of the SPM is a module that plugs into the bus and encompasses all the functionally required by this specification. The purpose of the VMIU is to provide a facility for mediation of CPU memory references without the necessity of an intermediate bus cycle for delivery to the SPM module. This implementation provides the potential for significantly reducing the performance degradation imposed by security.

3.1.2 SPM Interfaces

The SPM enforces security through mediation of all communication between the non-secure hardware components.

The interfaces are:

Processor to Memory

Device to Memory

3.1.2 SPM Interfaces (Continued)

Processor to Device

Device to Processor

Processor to Processor

3.1.2.1 Processor to Memory Interface

3.1.2.1.1 Address Translation

The SPM shall mediate all processor to memory references. When the processor makes a memory reference, the memory address is intercepted by the SPM and is treated as a virtual address. The SPM translates this virtual address into a physical memory address through a series of look-ups in descriptor tables resident in memory. The physical address is then presented to memory, and the appropriate read or write access is made. The data going to or from memory is not examined by the SPM.

Each memory descriptor in the descriptor tables contains, among various control fields (see Section 3.1.2.1.3.1), a pointer to an absolute memory location (i.e., a physical memory address). There are several types of descriptors, as designated by particular encodings in the descriptor control fields. If the descriptor is indirect, the descriptor's pointer is the address of another descriptor table. If the descriptor is direct, the object described is either and area of memory or an I/O device. If an area of memory, the descriptor's pointer is the address of a block of data to be referenced. This section will discuss in detail indirect and direct memory descriptors. See 3.1.2.3.3 for

a discussion of I/O descriptors.

The virtual address presented by the processor can, in the general case, be considered to consist of four fields, designated A, B, C, D, as shown at the top of Figure 3. The translation of a virtual address into a physical address as illustrated in the figure shall proceed as follows:

1. The SPM, given a virtual address, makes its first reference to the first level descriptor table pointed to by the descriptor base root (DBR) known to the SPM (see 3.1.2.1.3.2 for a discussion of the DBR).
2. The offset into this descriptor table is the first field of the virtual address (A), and the descriptor at that location is referenced.
3. If the descriptor is an indirect descriptor, the pointer in that descriptor is used to access a second descriptor table, and the second part of the virtual address (B) is used as an offset into this second table.
4. If the second level descriptor is indirect, it similarly is used to access a third descriptor table and the third part of the virtual address (C) is used to get the third level descriptor.
5. The third level descriptor must be a direct descriptor. Its pointer is used to find the page of data, and the last part of the virtual address (D) is an offset into the page to obtain the action word being referenced.

3.1.2.1.1 Address Translation (Continued)

The three-level descriptor system is the most general in that it allows for the implementation of segments, pages, and paged descriptor segments. The first descriptor table can be considered to be the page table of the descriptor segment, the second table is a page of the descriptor segment, and the third table is the page table for the segment. The indirect descriptors in the descriptor segment are called segment descriptors and the direct descriptors in the page tables are call page descriptors.

A process's view of memory is that of a series of segments, each identified by a Normal Segment Number (NSN) (composed of fields A, B combined). Within each segment there is a word offset (composed of fields C, D). Since each segment may not be the maximum size, there will be "holes" in the virtual address space for high values of the word offset (C, D) for some segments. Within a segment, however, all values of the word offset from 0 to the current size of the segment are usually defined.

Another variation on the address interpretation shall be implemented to allow unpaged descriptor segments. If, in a given application, it is determined that a process's descriptor segment will be no greater than one page, or that it is not necessary to page descriptor segments, it is useful to specify that the descriptor must be directly accessed by the offset specified in the combined A, B field

3.1.2.1.1 Address Translation (Continued)

(see Figure 4). In this case, the DBR points directly to the second level descriptor table, and the combined A, B field is used to index into this table. The T field in the DBR specifies this form of DBR interpretation (see 3.1.2.1.3.2).

One final variation shall be implemented to allow unpaged data segments which use the combined C, D field as an index. Unpaged data segments may be used with either paged or unpaged descriptor segments.

The virtual address field presented by the Level 6 CPU over the bus to the SPM is shown in Figure 5A. The field is 24 bits; however, the virtual address is restricted to 2^{20} words of 2^{21} bytes. A word consists of 2 bytes and a byte is an 8-bit information unit. The SPM shall be designed to accept and map a virtual address of 20 bits. The byte bit shall be passed on unchanged. The most significant bit of the 24 bit address field shall indicate that this is a virtual address and is to be mapped by the SPM. If this bit is a "0", it shall indicate that this address has been successfully mapped by the VMIU and requires no mediation by the SPM.

The different views of memory as seen by a process are shown in Figure 5B. The process can see up to 512 discrete segments of 2048 words each. In the case where the DBR is direct, the nine high-order bits are interpreted as a

3.1.2.1.1 Address Translation (Continued)

segment number and are used to index into a segment table to obtain a descriptor. If the descriptor obtained is direct, the low-order 11 bits are used as an offset into a 2048-word segment. If the descriptor is indirect, the next four most significant bits are interpreted as a page number and with two low order bits concatenated are used to index into a page table to obtain another descriptor. If the second descriptor is direct, the remaining 7 bits are used as an offset into a 128-word page. If the second descriptor is indirect, a trap back to the CPU is generated.

In the case where the DBR is indirect, the most significant 4 bits of the virtual address are used to index into a table to get the first descriptor. If this first descriptor is direct, a trap back to the CPU is generated. If the first descriptor is indirect, the next five most significant bits are used to index into a table to obtain a second descriptor. If this descriptor is direct, the least significant 11 bits are used as an offset into a 2048-word segment. If the second descriptor is indirect, the next 4 bits are used to index a page table to fetch a third descriptor. If the third descriptor is direct, the least significant 7 bits are used as an offset into a 128-word page. If the third descriptor is indirect, a trap back to the CPU is generated.

3.1.2.1.2 Access Control

In addition to performing the function of address translation, the SPM shall verify that the process has the required access to the memory location referenced. Access to a memory location is defined to be in one of the three modes: read (R), write (W) or execute (E). Read refers to a data or address constant fetch from memory, write is a store into memory, and execute is an instruction fetch from memory. There is a set of three ring brackets (R1, R2, R3) that are also used to determine the type of access allowed. The ring brackets restrict the process to certain types of access when executing in a given domain, or ring. Each memory descriptor shall be capable of containing the access permission and ring bracket information that is to apply to the location referenced. During the address translation phase, the access control information in the appropriate descriptor is used to calculate the final effective access mode to the location in memory. The effective mode is compared to the desired access mode, and an access violation trap shall be signaled by the SPM if the required access is not allowed by the effective mode.

Since, during any memory reference there are up to three descriptors that are accessed, a decision has to be made as to where to put the access control information. The general three level descriptor structure shown in Figure 3 requires that the access control information be placed in the second level descriptor, i.e., the segment descriptor.

3.1.2.1.2 Access Control (Continued)

This is because access to a segment may not be the same for every process currently using the segment. If the access control information were in the direct page descriptor, all processes using the page descriptor (which is shared) are forced to have the same access. If the access control information were placed in a page descriptor for the descriptor segment, the granularity of access control would be on the order of many segments.

There are specific cases, however, such as the use of unpaged or unshared segments, in which it is convenient to place the access control information in direct descriptors. Thus, in order to support full generality, the SPM shall be prepared to accept control information from any descriptor encountered during address translation. A field in the descriptor shall specify that the access control information it contains is to be applied to the memory reference. It is the responsibility of the security kernel to properly set the access control bits in each descriptor. More discussion of access control can be found in 3.1.2.1.3.1.

3.1.2.1.2.1 Effective Ring (Reff)

The actual effective access to a location in memory shall be determined by comparing a calculated effective ring number, Reff, to the three ring brackets associated with a descriptor for that memory reference, and then factoring in the three access permission bits. See 3.1.2.1.3.1 for a description of the exact algorithm used to calculate the effective mode.

3.1.2.1. Effective Ring (Reff) (Continued)
2.1

For the simple memory reference, the value of Reff used in this determination is the current ring number (Rcur) maintained by the CPU. In the general case, however, as part of the address preparation cycle, the processor may make a memory reference to fetch an indirect address (address constant) before operand fetch. (The fetch of an address constant from memory is subject to the same access control and address translation as a simple read access to data.) If an address constant is contained in a segment that can be written from a higher ring than Rcur, as is the case when an inner ring procedure is referencing arguments through an indirect address passed to it from an outer ring, the ultimate location referenced by the address constant must be subject to access control defined by the ring of the segment in which the address constant resides, rather than Rcur. If the address constant were only subject to Rcur restrictions, the inner ring procedure would, in software, have to verify that the address constant pointed to a segment to which the outer ring had access. In order to support software validation of arguments, the SPM shall validate the reference with respect to the ring of the segment in which the address constant resides.

The SPM shall accomplish the automatic address validation by keeping track, in terms of Reff, the maximum value of the ring number R1 in all descriptors encountered during address preparation. The value of Reff shall be initialized

3.1.2.1. Effective Ring (Reff) (Continued)
2.1

to Rcur at the beginning of each instruction cycle and shall apply to the instruction fetch and all references until the next instruction fetch. For each descriptor encountered between instruction fetch and operand fetch, a new value of Reff shall be computed as the maximum of the current Reff and R1 in the descriptor and this new Reff shall apply to the fetch of subsequent indirect addresses or data. It can be seen from this scheme that Reff can only increase from its initial value of Rcur.

3.1.2.1. Descriptors
3

Every resouce that is allocated to a process shall be represented by descriptors. Descriptors are constructed by the security kernel and are structured in memory for use by the SPM. The descriptor structure is the prime data base for the state of allocation of the system resources. Copies of descriptors in use in the SPM are only valid if they reflect the memory originals. This section will specify the format and semantics of a memory descriptor and a Descriptor Base Root. I/O descriptors are specified in Section 3.1.2.3.3, I/O Descriptors.

3.1.2.1. Memory Descriptor
3.1

The normal memory descriptor recognized by the SPM is a four word descriptor. The format is shown in Figure 6. This section specifies the information required to be contained in a descriptor. Each piece of required information is identified and its purpose identified.

3.1.2.1. Memory Descriptor (Continued)
3.1

DT - Directed Trap: This field provides for software directed hardware traps. Two bits (four encodings) must be provided, one of which (10_2) does not cause a directed trap. All other values shall cause an SPM generated trap.

Access Control: Four items of information are defined: the A field, the Ring Brackets, Permissions, and the Wire Bit. The A (Access) field determines whether the access control fields of the descriptor are to be used to control access to all resources described by the descriptor (regardless of the number of subsequent levels of address translation). Two values must be provided: if the A field is ON, then this descriptor's access control fields apply; if OFF, either an inferior or superior descriptor must provide the necessary access control. If more than one descriptor is encountered during address translation, with the A field ON, the first descriptor (defining the largest resource) with the A field ON defines the appropriate access control. Of course, at least one descriptor with the A field on must be found. If the SPM does not find one, it will generate a trap.

The R1, R2, and R3 fields define the privilege rings. Each field shall contain at least four values (integers: 0, 1, 2, 3) so that the system supports at least four rings of access privilege. The interpretation of these fields is described below.

3.1.2.1. Memory Descriptor (Continued)
3.1

The Read, Write, and Execute (R, E, and W) fields define allowed modes of access to the described resource. Each field must have two values (ON and OFF): if ON, the respective mode of access is allowed; if OFF, the respective mode of access shall be denied.

The following rules specify the required interpretation of the above access control information. The item Reff is the effective ring number computed by the SPM during effective address formation (ref. Section 3.1.2.1.2.1, Effective Ring).

1. Write permission if and only if (W = ON) and (Reff $\leq$ R1);
2. Read permission if and only if (R = ON) and Ref $\leq$ R2);
3. Execute permission if and only if (E = ON) and (R1 $\leq$ Reff $\leq$ R2); (Via signals from the processor, Section 3.7.2.1.2, the SPM can distinguish instruction and data fetches. Rule 3 shall apply for instruction fetches and rule 2 shall apply for data fetches.)
4. The use of R3 and the precise rules for entry/return to/from a procedure resource are specified in Section 3.1.2.1.6, Cross Ring Movement. In general, Call permission if and only if (E = ON) and (R1 $\leq$ Reff $\leq$ R3);

Certain sequences of ring numbers are termed brackets to denote a range of allowed rings in which certain modes of access are possible. The term write bracket shall apply to rings 0 to R1, inclusive. The term execute bracket

shall apply to rings R1 to R2, inclusive. The term call bracket shall apply to rings R1 to R3, inclusive.

The wire bit (Y) is set by kernel software to indicate that the segment is in main memory. The SPM shall verify that the wire bit is on prior to initiating a I/O transfer to or from the memory. If off, the SPM shall trap.

Usage: The U, M, and fields record and limit the usage of the described resource. The U field has two values (ON and OFF): if OFF and the resource is accessed (in any mode: read, write, or execute), the SPM shall update the value to ON. The M field has two values (ON and OFF): if OFF and the resource is accessed in the write mode, the SPM shall update the value to ON. The C field controls the entry of elements of the described resource into a data cache. It has two values: if ON, the described resource may enter the cache; if OFF, the resource shall not be placed in cache storage.

Descriptor Type: The T field identifies the type of the descriptor. This field shall contain three bits. One encoding $(100)_2$ is interpreted by the SPM to say that the descriptor is describing a process segment directly. (The selected encoding will be represented in this document by the notation "DIR".) Another encoding $(001)_2$ is interpreted to say the descriptor is describing an array of segment descriptors. (This encoding will be represented in

this document by the notation "IND".) A third encoding $(010)_2$ will specify that the descriptor is a page descriptor. All other encodings are reserved for future use.

Base: The base field supplies the physical address of the base (in memory) of the resource described. The base field for indirect descriptors and DBRs shall be 16 bits and the SPM shall concatenate 4 low order zero bits to form the full address. Direct descriptors shall have a base address field of 13 bits with 7 low order zero bits concatenated by the SPM.

Limit: The L field defines the size of the defined resource. An access request having an offset greater than the value of the L field of any descriptor encountered during address formation shall cause the SPM to generate a trap. The L field shall be 11 bits which provides the capability of specifying the maximum size of a resource as a single memory location.

Concurrent Access: The IOCT field of segment descriptors shall be incremented by the SPM at each initiation and decremented at the completion of an I/O operation in/out of the described resource. The field is intended to be used by system software to determine the existence of I/O operations in progress within a resource. This information shall then be used, by system software, to keep the resource in memory until all outstanding I/O has completed.

3.1.2.1. Memory Descriptor (Continued)
3.1

If the requested mode of access, for a resource, is not permitted by the access control information, the SPM shall generate a trap.

The interpretation of descriptor fields is dependent on the descriptor level (ref. Section 3.1.2.1.1, Address Translation). The T, C, DT, BASE and L fields are applicable for each level of descriptor. The U and M fields are referenced and updated only for direct descriptors. The access control fields A, R1, R2, R3, R, E, W, and Y are only applicable for a descriptor which has the A field ON. The IOCT field is applicable to segment descriptors only.

3.2.1.2. Descriptors
3.2

A special form of memory descriptor is recognized by the SPM. This descriptor is called the Descriptor Base Root (DBR) and is shown in Figure 7. It is used by the SPM to establish the set of descriptors for a process. The DBR is a 4 word construct similar in format to a memory descriptor. The first two words establish the set of memory descriptors. The second two words establish the set of I/O descriptors, (reference Section 3.1.2.3.3). The interpretation of the fields of the DBR by the SPM will be:

BASEM - Four low order zeros are concatenated to form the absolute location of the base of the element described, the root of the memory descriptor tree.

3.1.2.1. Descriptors (Continued)
3.2

RFU - Reserved Future Use.

T-Type Field - Encoding "DIR" implies that the set of memory descriptors is directly described. Encoding "IND" implies that a set of descriptors of the memory descriptors is described.

LIMIT - Limits the element described.

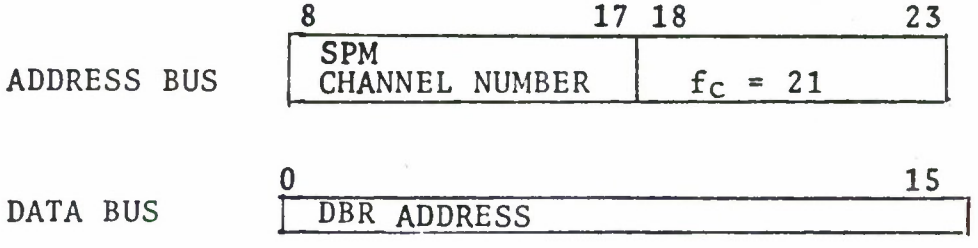
BASEI - The absolute location of the base of the I/O descriptor tree. Four low order zeros are concatenated.

3.1.2.1. Descriptor Structure Dynamics
4

3.1.2.1. Dispatch
4.1

The computer contains a dispatch function used for the initiation of a new process. This function notifies the SPM that a new descriptor structure is to be utilized for access mediation and the previous descriptor structure is to be discarded.

The dispatch function is an I/O output command, function Code 21, to the SPM that contains the absolute address of the new process DBR on the data bus.



3.1.2.1. Dispatch (Continued)
4.1

Upon receipt of the I/O Command, the SPM shall:

- Issue a WAIT response to the bus cycle and block the CPU.
- Upon detection of the SPM channel number, Ring 0, and the dispatch function code, the SPM shall concatenate two low order zeros to the address provided on the data bus and shall use this address to fetch and store the DBR for the new process.
- Concurrently, the SPM shall invalidate all descriptors contained in its cache except memory descriptors for I/O.
- The processor shall then be unblocked and the SPM shall issue an ACK to the repeated Dispatch order.

3.1.2.1. Selective Descriptor Invalidate
4.2

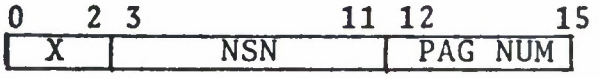
Changes made to the descriptor structure in memory for the currently executing process must be reflected in the copies contained in the SPM cache. This shall be accomplished via selective descriptor invalidation I/O output commands issued to the SPM. The SPM upon detection of the appropriate channel number and function code shall invalidate the selected descriptor(s). Subsequent resource accesses will then result in the SPM fetching the updated descriptor. The SPM shall support the following descriptor invalidation orders:

- I/O Descriptor invalidate, fc = 27, invalidate all I/O descriptors.

3.1.2.1. Selective Descriptor Invalidate (Continued)
4.2

- Selective Segment Descriptor Invalidate, fc = 29, invalidate the segment and all page descriptors for the virtual NSN on the data bus.

- Selective Page Descriptor Invalidate, fc = 2B, invalidate the memory page descriptor for the virtual NSN and PAG NUM on the data bus.



- Selective I/O Memory Descriptor Invalidate, fc = 2D, invalidate the I/O memory descriptor for the absolute channel number on the data bus. Decrement the IOCT field in the descriptor for the segment of memory involved in the I/O operation.

3.1.2.1.5 Cross Ring Movements

The CPU maintains a current ring number (Rcur) at which the processor is running. This ring number is used by the SPM in the calculation of the effective ring number (Reff) associated with a particular reference to memory that is compared to the ring brackets (R1, R2, and R3) of the referenced segment. Ring changes are initiated at the request of the process using the call and return instructions, or automatically by a trap or interrupt. This section discusses the call, return, and trap requirements. Interrupts are discussed in 3.1.2.4.1.

3.1.2.1.5.1 Call and Return

Two processor orders that shall be recognized by the SPM are the call and return orders. The call order is very similar to a transfer except that the SPM can change the current ring number to a lower value. The return is also a transfer with a possible increase in the current ring number. Calls are normally used to transfer to inner ring procedures to accomplish more privileged operations than those allowed at the current ring, and returns are used to return from an inner ring procedure back to the outer ring from which the call originated.

Access checking on the operand of the call instruction is somewhat different from that of other instructions. The operand of a normal transfer instruction need not be accessed until the next instruction fetch cycle, and thus access to the operand may not be required or checked until

the program counter is loaded with the new virtual address generated by the transfer instruction. Since the call instruction can change Rcur to a lower number and thus put the processor in a more privileged state, the SPM must guarantee that entry into the inner ring is tightly and completely controlled by that inner ring. This means that the SPM must check that calls can only be made to specific locations within specific procedures belonging to the inner ring.

The mechanism that accomplishes this control shall be implemented as follows. An inner ring procedure that is callable from an outer ring is defined as a "gate" by specifying in the ring brackets of the descriptor for the procedure segment a value of R3 that is different from R2. Normally, transfers to a segment cannot be made from rings above R2. However, a call instruction is allowed to a procedure if the call is made from a ring less than or equal to R3. If such a call is made, the new value of Rcur becomes R2, and execution continues. The value of Reff after address preparation for the call instruction is used in the comparison with R2 and R3. The tests made in the call are as follows:

Reff > R3	entry denied, trap (outside call bracket)
R2 < Reff $\leq$ R3	entry allowed, R2 becomes Rcur
R1 $\leq$ Reff $\leq$ R2	entry allowed, Rcur unchanged
Reff < R1	entry denied, trap (outside call bracket)

3.1.2.1. Call and Return (Continued)
5.1

The checks on call shall not preclude using the call instruction to transfer to a procedure from within its executive bracket. Nor shall it be required that a segment be a gate (i.e., R2 < R3) in order to be called from within execute bracket. Thus, the call bracket is defined as R1 to R3, with R2 being the new ring of execution if the segment is a gate and the call is from outside R2.

It is not sufficient to simply specify which segments are gates. There must also be a mechanism for specifying the location in the gate segment that is the valid entry point. This shall be done by allowing only location zero of the resource defined by the segment descriptor to be a valid entry point. Thus, the SPM during the access check discussed above, shall verify that the offset of the virtual address is zero before changing the ring of execution.

Outward calls are prohibited because of the potential for compromise. The Return instruction is used for outward ring crossings.

The CPU shall deliver the virtual entry point address over the bus as a conventional memory reference. Accompanying the address will be a CALL instruction signal derived from the CPU/SPM private interface. The SPM shall use the memory descriptor tree structure to obtain a valid direct descriptor. However, unlike normal memory references, no mapping need be performed. Instead the SPM shall:

3.1.2.1. 5.1 Call and Return (Continued)

- Validate that the caller has execute access at the entry point address.
- Verify the entry point is location zero of the called procedure (NSN OST = 0).
- Compute a new value of Rcur as specified in this section if all checks pass, set the descriptor U bit to ON if OFF, and unblock the processor which, in the absence of a trap from the SPM, will allow the virtual entry point address to be inserted into the CPU program counter and the new value of Rcur into the CPU S register.

Another transfer instruction that shall be recognized by the SPM is the return instruction. The only requirements for return are that the returning procedure be able to specify the ring to which to return and that returns to inner rings be prohibited. The ring to which the procedure desires to return, Rto, is delivered from the CPU to the SPM during the return instruction. The SPM shall verify that Reff $\leq$ Rto. If Rto < Reff, the SPM shall generate a trap.

3.1.2.1. 5.2 Trap and Trap Return

Traps are software initiated events (either intentional or unintentional) to which the processor responds by saving the current state of the processor in such a way that it can later be restored, and transferring control to a specified memory location.

3.1.2.1. 5.2 Trap and Trap Return (Continued)

Upon the occurrence of a trap, CPU firmware shall select an area for storage of information about the state of the process and an entry point to a service procedure. Since the CPU provides a single storage area/entry point per trap type/trap occurrence and traps may occur within any ring, all trap storage agrea shall reside within the security kernel (i.e., ring 0).

Upon occurrence of a trap, the SPM shall force Reff = 0, and shall translate the hardware-generated virtual addresses which specify the trap handler entry point and the trap save area the same as conventional memory references. For SPM initiated traps, the SPM shall store the SPM fault registers in the trap save area using firmware addresses from the CPU.

The trap return instruction (RTT) shall restore the state of the process from the trap save area as modified by the associated service procedure. No special SPM checks are required during trap return.

3.1.2.1. 5.3 Processor Generated Addresses

A special class of addresses going to the SPM from the processor are dedicated addressses originating from the processor firmware. Some of these addresses are system wide: Real Time Clock (RTC), Watch Dog Timer (WDT). Other addresses, such as the trap and interrupt vectors and next available trap save area are process oriented. The SPM

3.1.2.1.5.3 Processor Generated Addresses (Continued)

shall detect all generated addresses and shall treat them as virtual addresses with ring 0 privilege.

3.1.2.2 Device to Memory Interface

3.1.2.2.1 I/O Flow

There are two alternative data paths from device to memory specified. Each device attached to a secure data communications processor shall use at least one. The basic difference between the alternatives is defined by the nature of the information resident in a DMA device, where a Direct Memory Access (DMA) device, once initiated, will control a series of data transfers to (from) memory.

The first type of device to memory mediation, premapped I/O, interprets and translates memory addresses at I/O initiation and the device subsequently uses absolute addresses. The alternative, mapped I/O, requires SPM mediation of each memory request by the device. The SPM shall handle both types of flow and at I/O initiation use information within the I/O device describing mechanism (ref. Section 3.1.2.3.2, I/O Descriptors) to determine which flow is applicable.

3.1.2.2.1.1 Premapped I/O Flow

The premapped I/O flow is shown in Figure 8. This figure is meant to illustrate the flow of the addresses associated with an I/O transfer. At premapped I/O initiation, the virtual address associated with the transfer is delivered to the SPM. After suitable checking (3.1.2.3.3), the address is mapped by the SPM to an absolute memory address

3.1.2.2. Premapped I/O Flow
1.1

and loaded into the device. Transfer of data will occur directly between the device and memory using absolute addresses. The SPM shall mark the segment descriptor by incrementing the IOCT field at I/O initiation time so that the system will know not to reallocate these memory locations during the I/O operation.

The SPM shall determine that a device is to be treated as a premapped I/O device by an examination of the MT bit of the I/O descriptor. For initiation of data transfer for Direct Memory Access (DMA), the SPM shall insure:

1. That the device has been assigned to the process as indicated by the presence of an I/O descriptor.
2. That all memory addresses affected by the transfer have been wired and the proper access permission for the effective ring number of the process requesting the transfer. This shall be accomplished by checking the memory descriptor access field.
3. That the range of affected memory addresses falls within the range of memory described by one direct memory descriptor. This shall be checked by comparing the virtual address offset plus the range against the limit field contained in the memory descriptor.
4. That the Descriptor defining the I/O device allows access in this mode at the effective ring number of the process requesting the transfer.

3.1.2.2. Premapped I/O Flow (Continued)
1.1

If any of these checks fail, the SPM shall initiate a Trap. If all checks pass, the SPM shall proceed to map the I/O channel number and the starting address. The SPM shall receive from the CPU, via the two bus cycles of the IOLD instruction, the virtual channel number, the virtual starting address, the range or number of words to be transferred, and a function code indicating read or write. See Figure 9. The SPM shall map the virtual channel number into an absolute channel number using the I/O descriptor. The SPM shall pass the range and function code unmodified. Via two bus cycles, the SPM shall send the absolute information to the device. See Figure 10. Transfer of data shall occur directly between the device and memory without any intervention by the SPM.

3.1.2.2. Mapped I/O Flow
1.2

The address flow for the mapped I/O flow is illustrated in Figure 11. At mapped I/O initiation, the virtual address associated with the transfer is delivered to the SPM, and then is loaded into the device as a virtual address. The address of each item of data transferred shall be delivered to the SPM for mapping and checking. Each address delivered to the SPM shall be accompanied by the indentification of the transferring device so that the correct memory descriptor may be obtained by the SPM. The SPM shall retain, for each active I/O device, the following information. (An active I/O device is one in which an initiated I/O operation has not yet terminated.)

3.1.2.2. Mapped I/O Flow (Continued)
1.2

1. The effective ring number the device is to operate at.
2. Some method by which the SPM may access the memory descriptors of the process that initiated the I/O operation. For example, the SPM may remember the DBR contents at the time the I/O operation was initiated.

Each access by the device, to memory, is to be mediated by the SPM. The access checking performed by the SPM is equivalent to the checking performed for memory accesses by a processor.

Each access is evaluated at the effective ring number of the device, in the mode of the device (read or write), using the descriptors contained in the address space of the process that initiated the I/O operation.

Since the SPM retains descriptors recently used by I/O devices in its Back-up Storage Cache, the SPM shall provide the capability to clear the BUSC selectively, by device.

All addresses from I/O devices on the bus are virtual memory addresses and will be mapped by the SPM prior to their use in addressing memory. All virtual addresses arriving at the SPM shall be accompanied by an identification of the requesting device so that the proper memory descriptor may be selected. The information that the device is mapped is contained in the MT bit of the I/O descriptor. For initiation of mapped data transfer for DMA, the SPM shall insure:

3.1.2.2. Mapped I/O Flow (Continued)
1.2

1. That the device has been assigned to the process, indicated by the presence of an I/O descriptor.
2. That the I/O descriptor defining the device allows access in this mode at the effective ring number of the process requesting the transfer. This shall be done by checking the permission field of the I/O descriptor per Section 3.1.2.3.3.
3. That the starting address memory descriptor has the Y bit on.

If any of these checks fail, the SPM shall initiate a Trap. If all checks pass, the SPM shall proceed to map the I/O channel number. The SPM shall receive from the CPU during an I/O load instruction the virtual channel number, the virtual starting address, the range, and a function code indicating read or write. See Figure 12. The SPM shall map the virtual channel number into an absolute channel number using the I/O descriptor. Using the starting address NSN, the SPM shall increment the IOCT field of the segment descriptor, find the direct memory descriptor, and store it in BUSC along with the effective ring number and the DBR of the process requesting the transfer at the location dedicated to that channel. The SPM shall tag the descriptor and ring number with the channel number. The SPM shall then set the most significant bits of the starting address equal to the channel number and the rest of the starting address to the offset.

3.1.2.2. Mapped I/O Flow (Continued)
1.2

The SPM shall pass the range and function code unmodified. Via two bus cycles, the SPM shall send the modified information to the device. See Figure 13. When the virtual address associated with each request from the device for data transfer arrives at the SPM, the SPM shall be able to retrieve the memory descriptor and effective ring number by the channel number contained in the virtual address. The checking by the SPM during a mapped I/O transfer shall be identical to the chekcing of a memory access by the CPU. The SPM shall support mapped I/O page crossings by using the DBR stored at device initiation.

3.1.2.2 Processor to Device Interface

3.1.2.3. I/O Address Translation
1

The SPM shall mediate all processor to I/O references. When the processor makes an I/O reference, the address presented on the bus in intercepted by the SPM and is treated as a virtual address. The SPM translates this virtual address into a physical I/O address through a series of look-ups in descriptor tables resident in memory. The physical address is then presented to the device, and the appropriate transfer is made.

Each I/O descriptor in the descriptor tables contains, among various control fields, a pointer to an absolute memory location or I/O device (i.e., a physical memory address or physical I/O device). There are several types of descriptors, as designated by particular encodings in

the descriptor control fields. If the descriptor is indirect, the descriptor's pointer is the address of another descriptor table. If the descriptor is direct, the object described is an I/O device. This section will discuss in detail indirect descriptors and direct I/O descriptors.

The virtual address presented by the processor can, in the general case, be considered to consist of two fields, designated A, B as shown at the top of Figure 15. The translation of a virtual address into a physical address as illustrated in the figure shall proceed as follows:

1. The SPM, given a virtual I/O address, makes its first reference to the first level descriptor table pointed to by the descriptor base rool (DBR) known to the SPM (see Section 3.1.2.1.3.2 for a discussion of the DRB).
2. The offset into this descriptor table is the first field of the virtual address (A), and the descriptor at that location is referenced.
3. If the descriptor is an indirect descriptor, the format is that of a memory descriptor, the pointer in that descriptor is used to access a second descriptor table, and the second part of the virtual address (B) is used as an offset into this second table.
4. The second level descriptor must be an I/O descriptor. Its pointer is the absolute channel number of the I/O device.

3.1.2.3.2 I/O Descriptor

I/O descriptor are contained in the tree of descriptors rooted in (located by) the DBR (Figure 7). The SPM shall obtain the appropriate descriptor when presented with a virtual device name by the process. The format of the I/O descriptor is diagrammed in Figure 15.

The normal I/O descriptor recognized by the SPM is a 4 word descriptor. The interpretation of the fields of the descriptor by the SPM will be:

DT - A fault direction field settable by reference monitor software and checked by the SPM. One encoding $(10)_2$ is a no fault condition; all other encodings will cause the SPM to fault.

R1, R2, R3 - Ring Brackets for the resource described. For access rules, see next item.

R1, R2, R3, R, W, E - In developing an I/O virtual address, an effective ring number is generated in a manner similar to the effective ring number of memory addressing.

1. Write (to device) permission = (W=on) and (Reff $\leq$ R1)
2. Read (from device) permission = (R=on) and (Reff $\leq$ R2)
3. Control permission = (E=on) and (Reff $\leq$ R3)

Any condition not covered above results in a trap.

T Type Field - Standard interpretation.

MT - Indicates if the device is mapped or premapped.

0 = Premapped 1 = Mapped

3.1.2.3.2 I/O Descriptor (Continued)

CHANNEL NUMBER - Absolute device channel number.

U - Used Bit. This bit is set on, if off, by the SPM if the resource is used.

M - Modified Bit. This bit is set on, if off, by the SPM if the resource is to be written into.

SPM Channel Number - identifies the SPM that is on the same bus as the device described in a multiprocessor configuration.

Function Table Base Address - the 13 most significant bits of the base address of the Function Code Table for the device type associated with the descriptor.

3.1.2.3.3 I/O Function Codes

Associated with each I/O command from the processor is a 6-bit function code. Function codes may designate output or input operations. By convention odd function codes designate output transfers (Write) while even function codes designate input transfer requests (Read).

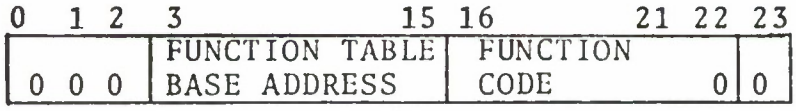
The SPM requirements for the mediation of an IOLD instruction are specified in Section 3.1.2.2. The function codes for the IOLD are fixed: 09 and OD hex. The SPM passes the function code without modification or checking on an IOLD transfer.

The SPM shall respond to an I/O input or output command by mapping the channel number and shall pass the function code unmodified if E = on and Reff $\leq$ R3.

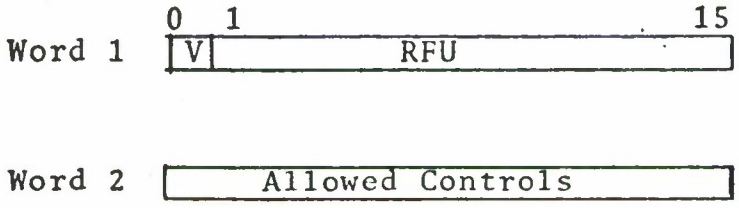
3.1.2.3.3 I/O Function Codes (Continued)

If E = off or Reff > R3, then the SPM shall verify that the operation being commanded is allowable by:

1. Concatenating the 6-bit function code to the 13-bit function table base address contained in Word 3 of the I/O descriptor and adding a least-significant zero to form the address shown.



2. Using the address thus formed to access two words.



The first word contains a validity bit and if V = 0, the SPM shall generate an invalid operation trap to the CPU. If V = 1, the SPM shall check that Word 2 contains a 1 in every bit position that the I/O data word contains a 1.

3. If the device and other checks pass, the SPM shall deliver the function code to the device unmodified.

3.1.2.3.4 SPM as a Device

3.1.2.3.4.1 Positional Priority

The bus has a positional priority via a distributed tie-breaking network that resolves simultaneous requests for bus cycles. In any system, memory is granted highest priority and the CPU has the lowest with other units being positioned on the basis of their performance requirements. Within the limiting physical constraints imposed by the cable connection to the CPU, the SPM is ideally located at a priority higher than mapped and lower than premapped devices.

3.1.2.3.4.2 Request Priorities

The SPM shall mediate requests in the order they are received and the process shall be uninterruptible.

If the CPU delivers a virtual address to the SPM over the bus, the SPM shall block the CPU from making any further requests. SPM requests from I/O devices shall be stored within the SPM until the current SPM transaction is completed. The SPM shall check and process all pending requests before unblocking the CPU. The SPM shall not store CPU requests. If there is no hit on the VMIU and the SPM is busy, the SPM shall issue a WAIT response to the CPU cycle.

3.1.2.3.4.3 SPM Bus Cycle Responses

Prior to initiating any CPU bus cycles, the address is sent to the VMIU for mediation. For any cycle not successfully mapped by the VMIU, the VMIU shall set the SPM flag bit

3.1.2.3. 4.3 SPM Bus Cycle Responses (Continued)

(address bit 0) to direct the pending bus cycle to the SPM for mediation. In addition, the VMIU shall set the memory reference signal (BSMREF-) true for any I/O bus requests. This will prevent any I/O device whose absolute channel number corresponds to the virtual channel number of the I/O request from responding.

The type of bus transfers that can occur and the four control line encodings that identify the type of transfers to the slave are shown in Table 1.

The SPM shall respond to CPU bus cycles in two distinctly different ways. In one category, the SPM can issue a response without first interrogating the slave. In the other category, the actual response of the distination must be obtained before the operation can be completed.

The SPM will respond on its own to the following types of bus cycles with an ACK or WAIT but never NAK:

1. Memory Read Request
2. Memory Write
3. Memory Write and Reset Lock
4. Memory Read Request and Reset Lock

In the case where the slave response can be either ACK, NAK, or WAIT, with equal probability of any one of the three occurring, the SPM shall respond with a WAIT to the CPU. The SPM shall then mediate the request, obtain the

BSMREF-	BSLOCK-	BSWRIT-	BSSHBC-	BUS CONTROL LINE / TYPE OF BUS TRANSFER
1	1	1	1	ABSOLUTE I/O INPUT REQUEST
1	1	1	0	I/O INPUT RESPONSE OR MEMORY READ RESPONSE
1	1	0	1	ABSOLUTE I/O OUTPUT OR INTERRUPT REQUEST
1	1	0	0	ILLEGAL
1	0	1	1	RFU
1	0	1	0	RFU
1	0	0	1	RFU
1	0	0	0	RFU
0	1	1	1	MEMORY READ REQUEST, VIRTUAL I/O INPUT REQUEST
0	1	1	0	ILLEGAL
0	1	0	1	MEMORY WRITE, VIRTUAL I/O OUTPUT REQUEST
0	1	0	0	ILLEGAL
0	0	1	1	MEMORY READ REQUEST, TEST & SET LOCK
0	0	1	0	MEMORY READ REQUEST, RESET LOCK
0	0	0	1	MEMORY WRITE, TEST & SET LOCK
0	0	0	0	MEMORY WRITE, RESET LOCK

TABLE 1. TYPES OF BUS TRANSFERS

3.1.2.3. 4.3 SPM Bus Cycle Responses (Continued)

destination response, and supply that response to the CPU reinitiated cycle. The following types of CPU bus cycles are in this category:

1. Virtual I/O INPUT REQUEST
2. Virtual I/O OUTPUT REQUEST
3. Memory Read, Test and Set Lock

In addition to bus cycles received from the CPU and I/O devices, the SPM can be the recipient of cycles from the memory or other SPMs in the system. The SPM shall respond with ACK, NAK, or WAIT as appropriate to the following bus cycles:

1. Memory Response
2. Absolute I/O INPUT REQUEST
3. Absolute I/O OUTPUT REQUEST

3.1.2.3. 4.4 SPM Function Codes

The SPM will be the recipient of both virtual and absolute I/O commands. All virtual I/O commands for the SPM originate from the associated CPU and if in Ring 0, the channel number shall be compared against the SPM channel number. If they compare, the SPM shall perform the operation specified by the function code as listed in Table 2. If not in Ring 0, the SPM shall mediate the request as defined in Section 3.1.2.3.

The SPM will receive absolute I/O commands from other SPMs in the system. NO checking or mapping is to be done by the

3.1.2.3. SPM Function Codes (Continued)
4.4

slave SPM. The SPM shall perform the operation specified by the function code as listed in Table 2.

TABLE 2

FUNCTION CODE (HEX)	OPERATION	SECTION REFERENCE
26	INPUT DEVICE ID	*
21	DISPATCH	3.1.2.1.4.1
27	INVALIDATE ALL I/O DESCRIPTORS	3.1.2.1.4.2
29	INVALIDATE SELECTIVE SEGMENT DESCRIPTOR	3.1.2.1.4.2
2B	INVALIDATE SELECTIVE PAGE DESCRIPTOR	3.1.2.1.4.2
2D	INVALIDATE SELECTIVE I/O MEMORY DESCRIPTOR	3.1.2.1.4.2

*
INPUT DEVICE ID; the SPM shall respond by inputing to the CPU, the SPM device ID. The SPM ID code is 2610 (hex).

3.1.2.4 Device to Processor Interface

The only device to processor interface is the signalling of interrupts by a device. The SPM does not mediate, receive or initiate interrupts. References by the CPU to the interrupt save area are treated by the SPM as standard generated addresses (Section 3.1.2.1.5.3). The interrupt return instruction, LEV, is executable by the security kernel only so the SPM need not perform any checks.

3.1.2.5 Processor to Processor Interface

In a system configured with multiple processors, each processor will work with its own SPM. Changes to the

3.1.2.5 Processor to Processor Interface (Continued)

descriptor structure will be made by the security kernel software, and in certain limited cases by SPM's. Changes to the descriptor by the SPM consist of marking the U and M bits and incrementing the IOCT field. The SPM shall use the memory lock line for an uninterruptible read-modify-write cycle when incrementing IOCT.

In a multiprocessor (multibus) system with the buses interconnected by an Inter-System Link (ISL), the SPM on the bus that has the I/O device associated with the transfer shall mediate the transfer. It is not required that this SPM be the initiating SPM; however, the descriptors to be encountered must be available to it.

The initiating SPM must deliver the DBR, Reff, absolute channel number, and memory descriptor to the mediating SPM.

The SPM channel numbers are restricted to 010 through 01F hex (000 through 00F are reserved for the processors). In a multiprocessor configuration, the SPM shall compare the 4 LSB of its absolute channel number against the SPM Channel Number field of the I/O descriptor during an IOLD order. If they compare the IOLD continues as normal. If they are different, the SPM must deliver the previously listed data to the SPM whose channel number is specified in the I/O descriptor.

3.1.2.6 Operator to Processor Interface

3.1.2.6.1 Standalone Bootstrap

When the secure data communications processor is to be operated in a standalone environment, some I/O device shall be controlled by the system operator to effect an initial memory load. In Figure 16 is shown the contents of memory following the initial memory load. This figure is meant to be illustrative, and is not intended to preclude other designs of the bootstrap mechanism. In Figure 16, a DBR, two I/O descriptors, two memory descriptors and a procedure segment have been loaded. The DBR establishes the trees of I/O (2) descriptors and memory (2) descriptors. The first I/O descriptor establishes the SPM as a device, the second established a device for further memory loading. The first memory descriptor establishes the loaded procedure, the second establishes a memory area for further I/O input. It is assumed that the processor Program Counter can be set to extract the first order of the procedure segment. The DBR is initialized either externally or by convention, by the bootload function, to a predefined value. The current ring is initialized to be zero. The contents of the Program Counter is assumed to be a virtual address and the corresponding instruction is fetched from memory using the initial DBR and memory descriptors. Processing continues in ring 0 (until explicity changed by software) with all addresses interpreted as virtual address.

3.1.2.6.2 Front-End Bootstrap

When the secure data communications processor is used as a front end for some host processor, it shall have the ability

3.1.2.6.2 Front-End Bootstrap (Continued)

to be bootstrapped from the host processor. Within the illustrative protocol of Figure 16, the initial memory load would be performed by the host processor through an interconnecting unit.

3.1.3 Major Component List

3.1.3.1 SPM

The SPM Block Diagram is shown in Figure 17. The SPM shall consist of the following major components:

SPM Hardware Interface
Virtual Address Holding Registers
Data Holding Registers
Control Holding Registers
Virtual Address Storage Memory
Data Storage Memory
Control Storage Memory
Back-Up Storage CACHE

Effective Ring Number Register
Adder
Limit Check
Permission Check
Absolute Address Holding Registers

Back-Up Comparator
Fault Register
Timing and Control

3.1.3.2 VMIU

The VMIU Block Diagram is shown in Figure 18. The VMIU shall consist of the following major components:

Descriptor Storage
Comparator
Adder Select
Adder
Limit Check
Permission Check
Firmware Detect

3.2 SPM Characteristics

3.2.1 Performance

Performance degradation of the system with the SPM shall not exceed 25% relative to a system without the SPM.

3.2.2 Physical Characteristics

The SPM shall tentatively consist of two (2) fifteen by sixteen inch (15" x 16") rectangular circuit boards. These boards will be of multilayer or double-sided construction, on 1.0 inch centers, to be contained within a ruggedized cast chassis. The SPM shall also consist of one (1) or more daughter boards. At least one (1) of these daughter boards (VMIU) shall be mounted on the CPU motherboard.

In accordance with the specific program environmental confines, an option will be offered which will mechanically stiffen the boards as a means for ruggedization. In either case, the boards will be rail-mounted within the chassis. Electrical interface will be provided at the aft corners of each board via plug-in connectors. The weight of each board is estimated at between two to four pounds. Maintenance access will be from the front, after hinging of the control panel.

Each board comprising the SPM should be transported and/or stored in an individual protective container.

The chassis which contains the SPM shall be compatible with MIL-STD-461A electromagnetic requirements.

3.2.3 Reliability

Design considerations and parts selection shall be sufficient to assure that the equipment meets or exceeds its reliability requirements over its useful life.

3.2.3.1 Mean-Time-Between-Failures (MTBF)

The design goal calculated MTBF for the SPM shall be greater than 20,000 hours. Calculation procedures shall be in accordance with MIL-STD-756 and Appendix A of MIL-HDBK-217B.

Microcircuit failure rates to be employed in the calculations shall be as follows:

IC Device Type	Failure Rate (Per 10^6 Hours)
SSI, less than 20 gates	0.03
MSI, 20 - 100 gates	0.05
LSI, greater than 100 gates	0.1
Bipolar Memory, RAM	0.3
MOS memory, 4096 bit RAM	1.0

3.2.3.2 Probability of Failure Induced Security Compromise

As a design goal, the SPM shall exhibit a probability of less than 0.000001 per hour that hardware failure will result in the undetected loss of secure data protection functions. The probabilistic measure of security compromise shall be established by analysis using failure rate data as specified in Paragraph 3.2.3.1.

3.2.3.3 Useful Life

The useful life of the SPM shall be 10 years minimum when operated and maintained in accordance with the provisions of this specification.

3.2.4 Maintainability

Maintenance of the SPM unit shall be effected by hinging of the chassis control panel and subsequently replacing any faulty or malfunctioning board with a spare board.

Faulty boards may then be returned to the manufacturer for detailed piece-part/circuit repair. This implies that the customer maintains an adequate stock of replacement spares.

Cost-effectiveness trade-offs have indicated that such a scheme, where the customer performs simple diagnostics to determine the faulty board for replacement, usually results in minimum service contract costs to the manufacturer.

3.2.5 Environmental Conditions

Temperature - The SPM shall be operable within ambient temperatures ranging from 0°C to 50°C.

Vibration - For the SPM within a hard-mounted chassis, the capability shall be sinusoidal vibration of 2.0g's peak from 5 Hz to 2,000 Hz. When installed within an isolated chassis, the capability shall be extended to 10.0g's peak from 5 Hz to 2,000 Hz.

Shock - The SPM within an isolated chassis will be capable of withstanding a half-sine input pulse of 15.0g's peak for a duration of 11.0 milliseconds. In an isolated unit, the SPM shall be capable of withstanding pulses in accordance with MIL-S-901C, for lightweight equipment.

Altitude - The SPM shall be required to satisfactorily

3.2.5 Environmental Conditions (Continued)

operate from 0 to 8,000 feet altitude.

Humidity - The SPM must satisfactorily withstand a relative humidity of 100 percent.

3.2.6 Transportability

Each board of the SPM shall be suitably packaged in its own protective container. Several boards may be shipped in the same container, provided that such is partitioned between component boards and that each board is individually foam-packed. Packing shall be sufficient to prevent damage to a board or board components in the event that the overall transportation container becomes damaged.

3.3 Design and Construction

3.3.1 Materials, Processes and Parts

Materials, parts and processes shall conform to the requirements of MIL-E-5400 when practical or unless otherwise restricted herein. Design and application considerations, as well as economic factors, shall govern the selection of and use of materials, parts and processes. HIS materials, parts, processes and controlling specifications used for the existing NML design and Aero FMS's and FPS's used for the SPM design shall be considered approved for the SPM upon verification of data substantiating that the unit will perform satisfactorily in the specified environment. In addition, the following paragraphs identify specific requirements and limitations in the use of materials, parts and processes.

3.3.1.1 Elastomeric Materials

Elastomeric components shall utilize only those elastomers which have adequate resistance to aging, ozone, heat aging, low temperature embrittlement and reversion, either temperature or moisture-temperature induced.

3.3.1.2 Wire

Wire used in all RNML new designs shall conform to the following specifications:

A. 300V, Single Conductor - FMS 40052

B. 300V, Shielded - FMS 40022

C. 600V, Single Conductor - FMS 40053

D. 600V, Shielded - FMS 40051

3.3.1.3 Conformal Coatings

Printed circuit cards shall be conformally coated per FPS 18035, Type V.

3.3.1.4 Processes

3.3.1.5 Soldering

Electrical soldering practices shall be in accordance with FPS 18167. Certification of soldering operators is required.

3.3.1.6 Parts Selection and Standardization

Electronic part types for all new SPM circuit designs shall be selected from the BCO Standard Parts List. Selection, qualification and screening criteria applicable to non-standard parts shall be in accordance with Parts Control Program Requirements of the Honeywell RNML Reliability Program Plan.

3.3.2 Electromagnetic Radiation

The SPM shall be designed in accordance with the guidelines contained within AFSC DH1-4, Electromagnetic Compatibility.

3.3.2.1 EMC

Electromagnetic compatibility criteria for the SPM when housed in the RNML chassis shall be in accordance with the emissions and susceptibility test requirements of MIL-STD-461A, Notice 3, 1 May 1970 for Class A3 equipment. Applicable requirements are:

CE03 - Conducted Emissions, power lines.

CE04 - Conducted Emissions, signal lines.

CS01 - Conducted Suscept., power lines, AF.

CS02 - Conducted Suscept., power lines, RF.

CS06 - Conducted Suscept., power lines, transient.

RE02 - Radiated Emissions, electric field.

RS03 - Radiated Suscept., magnetic induction field.

RS03 - Radiated Suscept., electric field.

3.3.2.2 TEMPEST

Tempest criteria for the SPM when housed in the RNML chassis shall be in accordance with the following portions of NACSEM 5100 as specified by DCA circular 370-D195-2:

Electric Field Space Radiation

Power Line Conduction

Black Signal Line Conduction

Red Signal Line Conduction

3.3.3 Nameplates and Product Marking

Identification and marking for Aero designed hardware shall be in accordance with MIL-STD-130. Identification and marking for BCO designed hardware shall be per BCO standards. If existing designs do not meet these requirements, it shall be documented and corrective action shall be taken if necessary.

3.3.4 Workmanship

Workmanship of the SPM shall be in accordance with the applicable portions of UED 23036. General workmanship shall be of high quality to assure compliance with specification requirements including the service life requirement.

3.3.5 Interchangeability

3.3.5.1 General

Mechanical and electrical interchangeability shall exist between like assemblies, subassemblies, and replaceable parts regardless of manufacturer or supplier. Interchangeability, as used here, does not mean identify, but requires that a substitute of like assemblies, subassemblies and replaceable parts may be easily effected without physical or electrical modifications to any part of the equipment or assemblies including cabling, wiring and mounting.

3.3.5.2 Module Interchangeability

Any one of the SPM's shall be replaceable and interchangeable without electrical adjustment or calibration.

3.3.6 Safety

The SPM shall be designed to combine maximum safety and stability, avoiding sharp edges, protrusions, obstructions and any other mechanical or physical features which constitute a hazard in accordance with MIL-STD-1472, Sections 5.13.4 and 5.13.5.

3.4 Documentation

3.4.1 Drawings

All Aero engineering released drawings shall be equivalent to or better than that required by MIL-STD-1000, Category E, Form 3. BCO drawings shall conform to HIS Standards.

3.4.2 Specifications

Specifications are required for all parts, materials and processes utilized in the fabrication and assembly of this unit. This requirement is necessary to assure the validity of Qualification Test Results.

3.4.3 Test Plans

Test plans shall be per Aero Design Procedures Paragraph 5.3.

3.5 Logistics

Maintenance procedures, supply, facilities, facility equipment, personnel, and training requirements shall be per the approved RNML Integrated Logistics Support Plan.

3.6 Personnel and Training

There are no Personnel and Training requirements relating to the SPM.

3.7 Major Component Characteristics

3.7.1 Security Protection Module (SPM)

The SPM shall provide the following processing and control functions.

3.7.1.1 SPM Hardware Interface

3.7.1.1.1 Bus Interface

The bus interface shall contain all the necessary circuitry to interface with the NML bus as specified in Honeywell Engineering Product Specification 60126298.

3.7.1.1.2 CPU/SPM Interface

The SPM/CPU shall have a private interface. This interface is via the VMIU and is defined in Section 3.7.2.1.

3.7.1.2 Virtual Address, Data and Control Holding Registers

The holding registers shall be capable of copying and holding information from the bus interface logic. They shall be loaded when the control signal, Data Coming Now (DCN), goes true unless the SPM is already busy mediating a previous request. The holding registers shall also be capable of being loaded from its associated storage memory by a control signal from the timing and control section. The combined holding registers shall be capable of holding all the information from two NML bus cycles.

3.7.1.3 Virtual Address, Data and Control Storage Memory

The storage memories shall be capable of copying and holding information from the bus interface logic. They shall be loaded when the control signal, DCN, goes true and the SPM is busy mediating a previous request. The

3.7.1.3 Virtual Address, Data and Control Storage Memory (Continued)

storage memories shall also be capable of being written into by the associated holding register upon command from the timing and control section. The address to be written into shall be supplied by an input address counter which will be incremented by one after each input to the memory. The storage memories shall be capable of being read out upon command by the timing and control section. The address to be read out shall be supplied by an output address counter which shall be incremented by one after each output from the memories. The storage memories shall contain enough bits to copy all the information from an NML bus cycle.

3.7.1.4 Back-Up Storage Cache (BUSC)

The SPM shall contain a Back-Up Storage Cache capable of holding descriptors, necessary parity and validity bits, and tags to describe each descriptor.

Two descriptor locations shall be reserved to hold the two Data Base Registers (DBR) for the process currently in control of the CPU.

At least one location shall be reserved to store the most recently used I/O descriptor(s).

Each system supports an I/O device naming structure (channel number) of 10 bits. The SPM shall support the entire naming structure. A minimum of 128 channels shall

3.7.1.4 Back-Up Storage Cache (BUSC) (Continued)

be supported by the SPM with memory descriptor storage in the BUSC. The remainder shall be supported by storage in main memory with firmware retrieval as required. The locations in BUSC shall be selected by using the LSBs of the absolute channel number.

A minimum of thirty-two (32) locations shall be used for memory descriptors. These locations may be subdivided into slots each providing multiple levels of descriptors. The goal of the subdivision and selection technique shall be to maximize the hit ratio.

3.7.1.5 Effective Ring Number Register

The Effective Ring Number Register shall contain the two bits of ring number that is the greater of the Current Ring Number and all R1 fields encountered in descriptors during virtual address transformation, except during instruction fetch. During instruction fetch, the effective ring number shall be equal to the current ring number. The Effective Ring Number Register shall be loaded with either the CPU current ring number or the R1 field of the descriptor from the BUSC.

During Interrupts and Faults, a special line from the CPU shall clear this register to zero in order to force Ring 0 during the save of the CPU registers.

3.7.1.6 Adder

The Adder shall add the absolute base address from the descriptor to the NSN or the page number of the offset under control of the timing and control section.

3.7.1.7 Limit Check

The Limit Check shall compare the Limit Field in the descriptor against the offset, or the NSN, or the page number under control of the timing and control section. If the Limit Field in the descriptor is the smaller of the two items compared, an error signal shall be generated and sent to timing and control.

3.7.1.8 Permission Check

The permission check shall perform all of the read, write and execute checks as specified in 3.1.2.1.3.1, 3.1.2.1.6.1 and 3.1.2.3.3.1. The permission check shall receive as inputs the read, write, execute and ring information from the descriptor store as well as the read, write or execute commands from the control holding register. In addition, the permission check shall receive the effective ring number from the effective ring number register. After performing all checks, the permission check shall indicate to the timing and control section if all checks pass. If all checks do not pass, the permission check shall set the appropriate bits of the fault register.

3.7.1.9 Absolute Address Holding Register

The absolute address holding register shall be capable of copying and holding the absolute address from the adder.

3.7.1.9 Absolute Address Holding Register (Continued)

It shall be loaded under control of the timing and control section. The holding register shall also be capable of being loaded from the absolute address storage memory by a control signal from timing and control. The register shall be capable of holding the entire absolute address.

3.7.1.10 Back-Up Comparator

The back-up comparator shall compare the tag stored in the back-up storage CACHE against the NSN, page or channel number from the virtual address holding register. The output of the comparator shall indicate if the inputs are equal or not.

3.7.1.11 Fault Register

The fault register shall store fault information generated by the SPM during mediation. The fault register shall contain 2 words of data and shall be capable of being stored in memory with addresses delivered from the CPU. As a minimum, the following information shall be stored in the fault register:

```
Bit 0 - 12:  13 MSB of Virtual Address
Bit 13:      Read Fault
Bit 14:      Execute Fault
Bit 15:      Write Fault
Bit 16:      Directed Trap Bit 0
Bit 17:      Directed Trap Bit 1
Bit 18:      I/O Fault
Bit 19:      Limit Fault
Bit 20:      Invalid Function Code

Bit 21 - 31: RFU
```

3.7.1.12 Timing and Control Section

The timing and control section shall generate all of the necessary signals to sequence the SPM through the mediation process.

3.7.2 VMIU

3.7.2.1 CPU/VMIU Interface

There are two physical interfaces between the CPU and the VMIU. The first is between the VMIU and the CPU motherboard. This interface delivers the virtual address to the VMIU, returns the mapped address to the bus logic and provides information and control lines required by the SPM. The other interface is between the CPU Register Arithmetic Logic Unit (RALU) daughterboard and the VMIU. This interface provides additional control and information lines between the CPU and VMIU.

3.7.2.1.1 VMIU/CPU Motherboard Interface

The following signals are available on this interface. Signals names suffized with a plus sign are true high signals.

Signal	Quantity	Function
MYAD03+ thru MYAD22+	20	Virtual address to VMIU.
GJAD03+ thru GJAD15+	13	13 most significant bits of mediated address from VMIU, concatenated with MYAD16-22 at CPU bus interface.
GJAD00+	1	SPM flag bit, set true by VMIU to virtualize VMIU address to bus.

3.7.2.1.1 VMIU/CPU Motherboard Interface (Continued)

Signal	Quantity	Function
RS01PF+10 RS02PF+10	2	Rcur from CPU, encoded per the following:

	RS01PF	RS02PF	
Kernel	1	1	Ring 0
	1	0	Ring 1
	0	1	Ring 2
	0	0	Ring 3

MCLOCK+	1	Clock at micro-frequency to VMIU.
MLRVLD+	1	Data settling blanking pulse, low 30 ns after clock leading edge to VMIU.
BUS CYC+	1	True for CPU initiated bus cycle to VMIU.
CIMREF+	1	True for memory reference bus cycle to VMIU.
CIREAD-	1	Read/write identifier to VMIU.
CIDBPL+	1	True for memory double fetch (instruction fetch) to VMIU.
GJDBPL+	1	True for memory double fetch from VMIU.
GJPROV+	1	Firmware testable line to CPU indicating SPM fault.
GJMREF+	1	Line held true by the VMIU to make all CPU initiated bus cycles memory references.

3.7.2.1.2 VMIU/RALU Interface

Signal	Quantity	Function
PLUPTB-	1	Open collector signal to CPU to stall MCLOCK high.
CMTMOT-	1	Open collector signal to initiate trap from SPM.
NAG002+ thru NAG011+	10	CPU control store address to detect firmware steps requiring special SPM actions.

3.7.2.1.2 VMIU/RALU Interface (Continued)

Signal	Quantity	Function
BIDBl1+ BIDBl2+	1 1	2 bi-directional bits of internal bus for transfer of Rto and Rcall between SPM and CPU.

BIDBl1+	BIDBl2+	
1	1	Ring 0
1	0	Ring 1
0	1	Ring 2
0	0	Ring 3

Signal	Quantity	Function
CRIFCI CRFUN0 CRFUN1 CRFUN2	1	4 encoded control signals per:

CRIFCI	CRFUN	1	2	
1	0	1	1	Start Instr. Fetch
1	1	0	1	End Instr. Fetch
1	1	1	X	Indirection
0	1	X	X	Set Reff = 0

3.7.2.2 Descriptor Storage

The VMIU shall contain a cache for storage of the VMIU required direct descriptor fields. The size and organization shall be such that the hit ratio is maximized within the physical and economic constraints.

3.7.2.3 Comparator

The comparator shall compare the tag from descriptor storage against the NSN and page from the virtual address presented by the CPU to the VMIU. The output of the comparator shall indicate if the inputs are equal or not equal.

3.7.2.4 Adder Select

The adder select shall select the offset to be added to the descriptor base. In the case of a direct segment

3.7.2.4 Adder Select (Continued)

descriptor, the adder select shall pass the eleven least significant bits of the virtual address. In the case of a direct page descriptor, the adder select shall pass the seven least significant bits of the virtual address.

3.7.2.5 Adder

The adder shall add the absolute base address from the descriptor to the offset from the adder select.

3.7.2.6 Limit Check

The limit check shall compare the limit field in the descriptor against the offset from the adder select. If the limit field in the descriptor is the smaller of the two items compared, an error signal shall be sent to the SPM.

3.7.2.7 Permission Check

The permission check shall perform all of the read, write and execute checks as specified in 3.1.2.1.3.1. The limit check shall receive as inputs the read, write and execute signals and the Reff signals. These inputs shall be checked against the permission field contained in the descriptor. If any permission checks fail, an error signal shall be sent to the SPM.

3.7.2.8 Firmware Detect

The firmware detect logic shall decode the CPU control store addresses to detect events such as CALL, RTN, and firmware generated addresses which require special action by the SPM.

4.0 QUALITY ASSURANCE PROVISIONS

4.1 General

The Quality Assurance Program to be applied to the SPM shall be conducted in accordance with the criteria described herein and the SCOMP Product Assurance Program Plan. The SCOMP P.A. Program Plan shall describe the integrated quality and reliability assurance activities applicable to SCOMP prototype and production systems.

4.1.1 Responsibility for Tests

Unless otherwise specified in procurement documentation, the supplier is responsible for the performance of all tests and inspections specified herein.

4.1.2 Special Tests and Examinations

The following requirements of Section 3.0 shall be verified entirely, or in part, by inspection of the equipment and its drawings:

A. (3.2.2) Physical Characteristics

B. (3.3.4) Identification and Marking

C. (3.3.5) Workmanship

D. (3.3.6) Interchangeability and Replaceability

4.1.3 Reliability Analysis

See paragraph 3.2.3.

4.2 Quality Conformance Inspections

4.2.1 Engineering Design Evaluation

4.2.1.1 Hardware Certification

A SCOMP logic design verification analysis shall be performed to verify that the SCOMP performance

4.2.1.1 Hardware Certification (Continued)

specifications are accomplished by the digital logic mechanization of the SPM in conjunction with the CPU. The analysis shall consist of two phases. First, development of correspondence between the SCOMP specification and this specification using hardware flow charts and a corresponding set of operating specifications which describe elements of SPM hardware performance in a simple way. The second phase of the analysis shall consist of detail logic analysis using register and/or instruction level simulation. The simulation shall include the SPM and portions of the CPU; other SCOMP elements may be analyzed manually.

Related to the hardware certification analysis task is an analysis of the probability that hardware failure will induce security compromise. This task is described in paragraph 3.2.3.2.

4.2.1.2 Design Evaluation Testing

4.2.1.2.1 Prototype Development Tests

A prototype SPM shall be subjected to design evaluation test sequences to verify its functionality and operation under worst case conditions of power temperature and clock frequency operation. The tests shall be conducted with the SPM installed in a minicomputer configuration whose functional elements have previously acceptance tested. SPM functionality shall be verified using operating software developed as specified in paragraph 4.2.1.2.2.

4.2.1.2.2 Prototype Test Software

The prototype SPM-SCOMP configuration shall be development tested using evaluation software developed with the aid of a CPU-SPM instruction simulator. Software developed on this simulator shall test the SPM mediation functions to insure that the performance requirements for the SPM described in paragraph 3.0 are exercised.

4.2.1.3 SPM Qualification Tests

Environmental qualification tests for the SPM are not required. Qualification for the SPM shall be established by structural similarity to ruggedized minicomputer circuit elements upon which tests shall be performed. The similarity units shall include at least one CPU and one 32K word memory.

4.2.2 Prototype Inspection and Test

SCOMP prototype subassemblies shall be visually inspected for workmanship, damage and assembly configuration prior to first powered operation.

Prototype SPM's shall be acceptance tested in accordance with paragraph 4.2.1.2.1.

4.2.3 Production Acceptance Tests and Inspections

4.2.3.1 Inspection Criteria

4.2.3.1.1 Workmanship

Workmanship shall be verified on each production SPM to Honeywell workmanship standard, OED 23036 to meet the requirements of MIL-STD-454 Requirement 9.

4.2.3.1.2 Configuration

Each production SPM shall be visually examined in individual parts kit form prior to issuance to assembly and again upon completion prior to acceptance testing. Configuration examination shall include:

- Verification that correct part types have been issued for manufacture.
- Completed assemblies are complete and visually identical to a standard reference SPM or photograph thereof.

4.2.3.1.3 Electronic Parts Inspection

The logic functionality, lack of damage and marking of integrated circuits to be assembled into production SPMs shall be verified by inspection and test prior to assembly. Appropriate quality control sampling plans based lot total percent defective (LTPD) acceptance criteria shall be employed for marking and damage.

4.2.3.2 Production Acceptance Testing

4.2.3.2.1 Acceptance Tests

Production acceptance tests shall be conducted under the supervision of quality control using approved test procedures, equipment and software. Each SPM shall be accepted with the SCOMP unit for which it is intended. Spare SPM's may be acceptance tested in any SCOMP if compatible configuration provided that all functional elements used in the test have been inspected in accordance with paragraph 4.2.3.1.

4.2.3.2.2 Production Test Software

Software used for acceptance testing of production SPM's shall be derived from the prototype software (see paragraph 4.2.1.2.2) or other suitable source which insures that each SPM mediation function is exercised.

Production test software shall be formally issued and controlled by quality assurance in accordance with Honeywell Design Procedure 3.3.

5.0 PREPARATION FOR DELIVERY

See paragraph 3.2.6.

6.0 NOTES

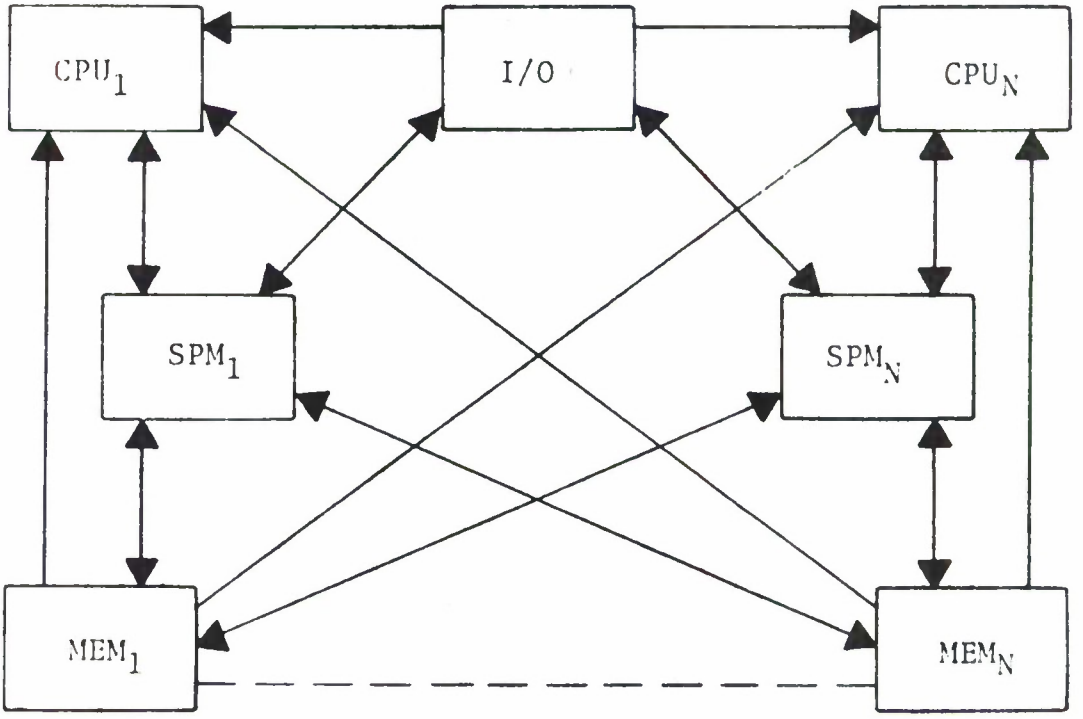


Figure 1. GENERAL SYSTEM STRUCTURE

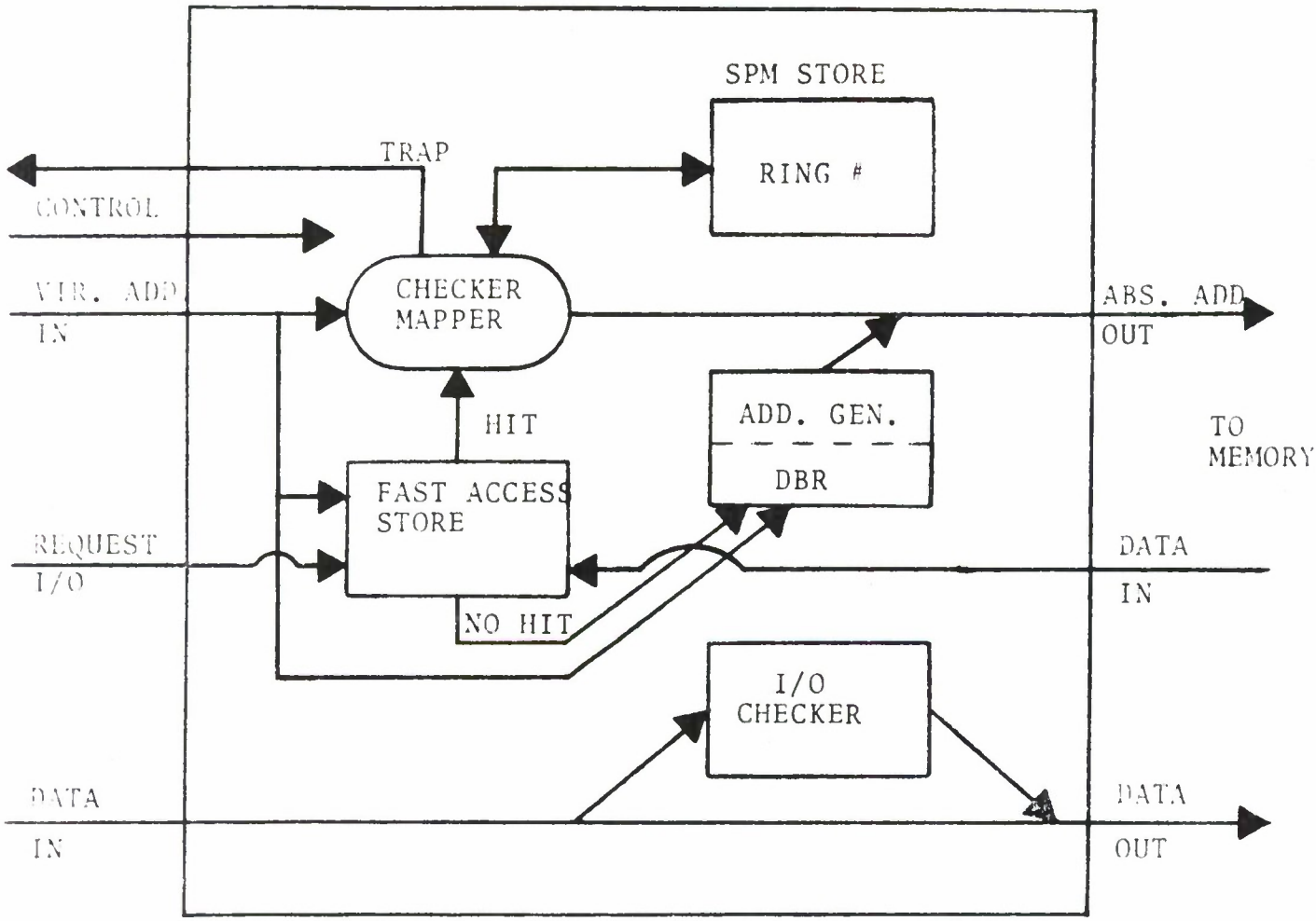


Figure 2. SPM FUNCTIONAL

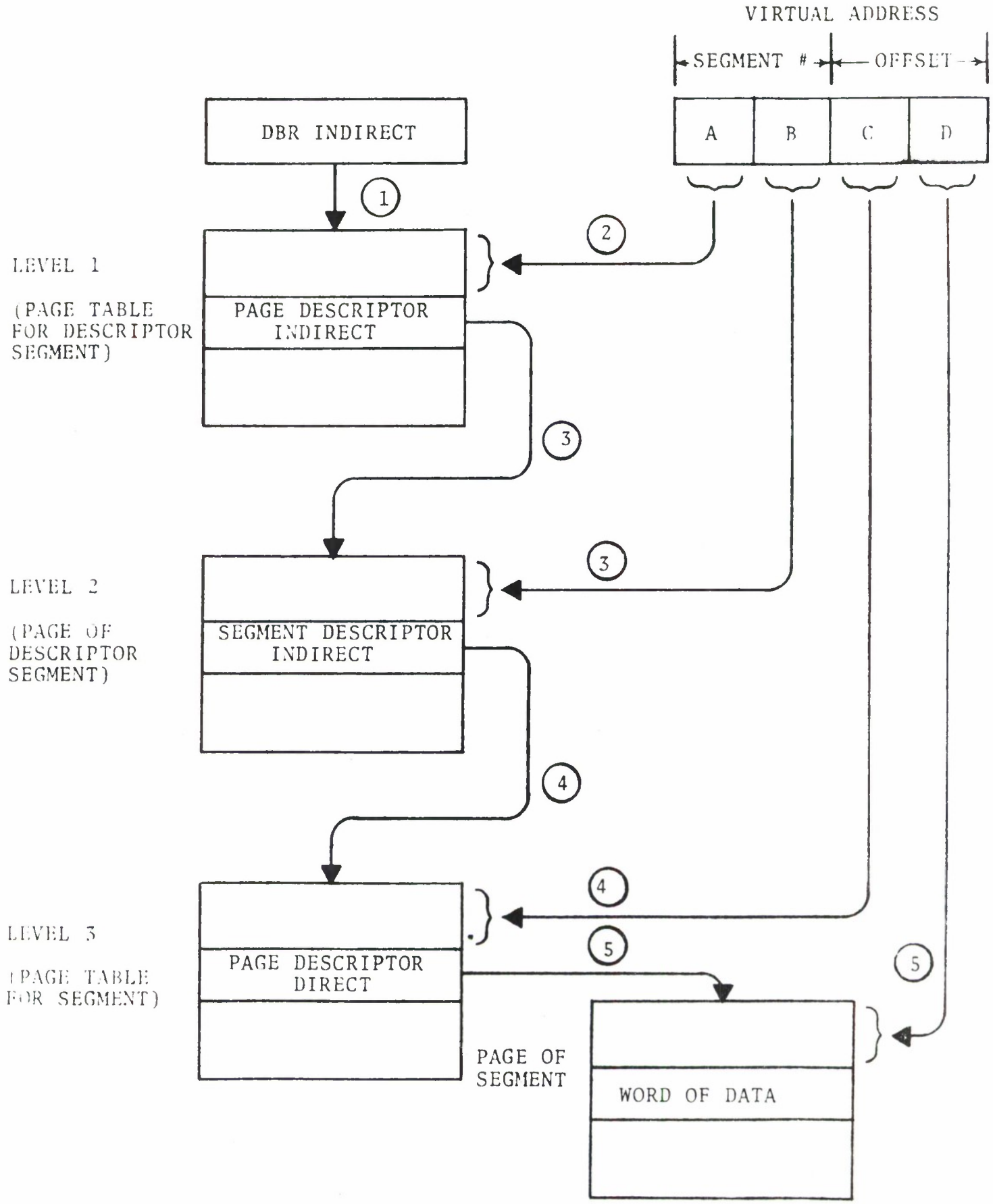


Figure 3. ADDRESS TRANSLATION

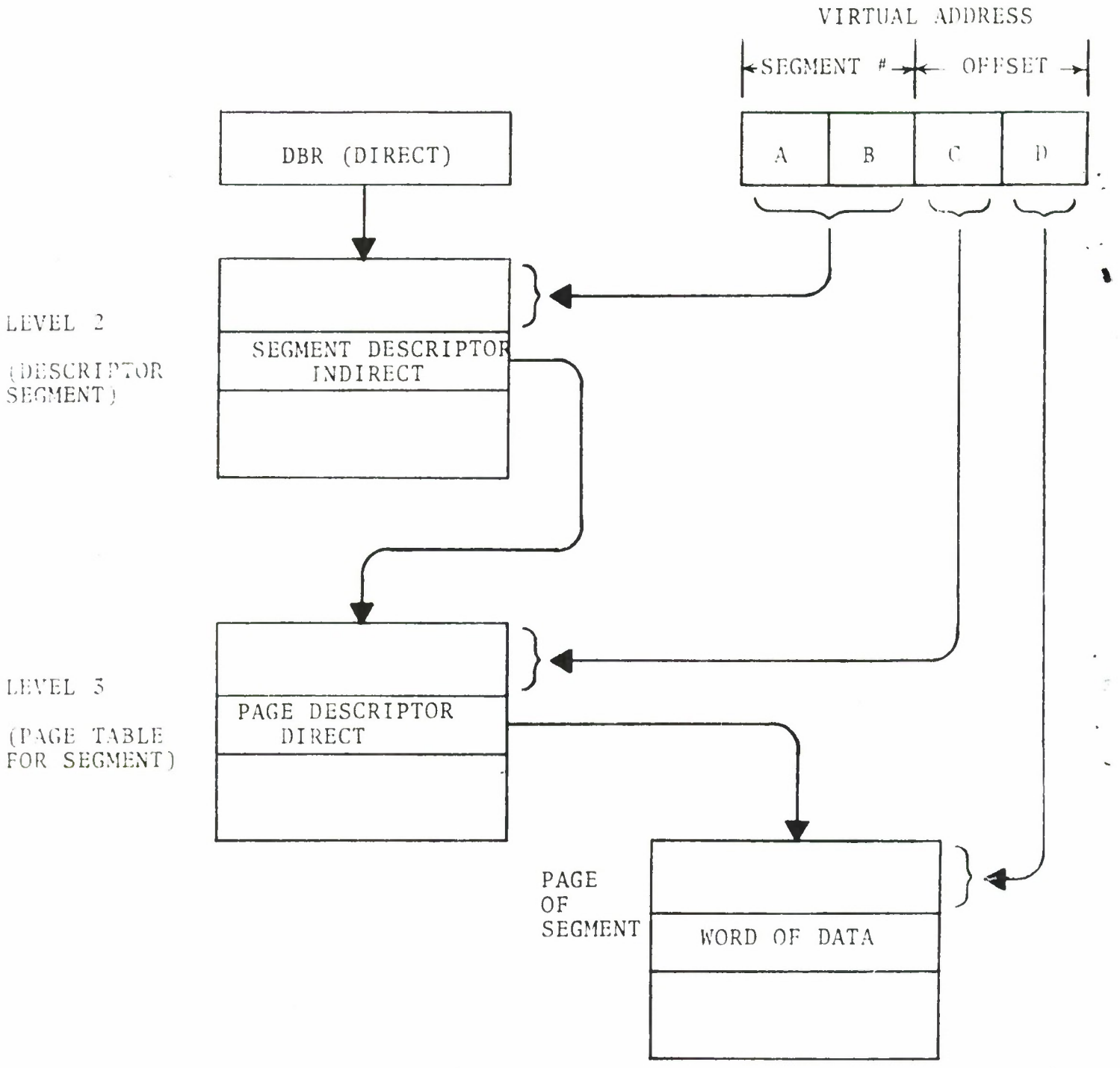


Figure 4. UNPAGED DESCRIPTOR SEGMENT

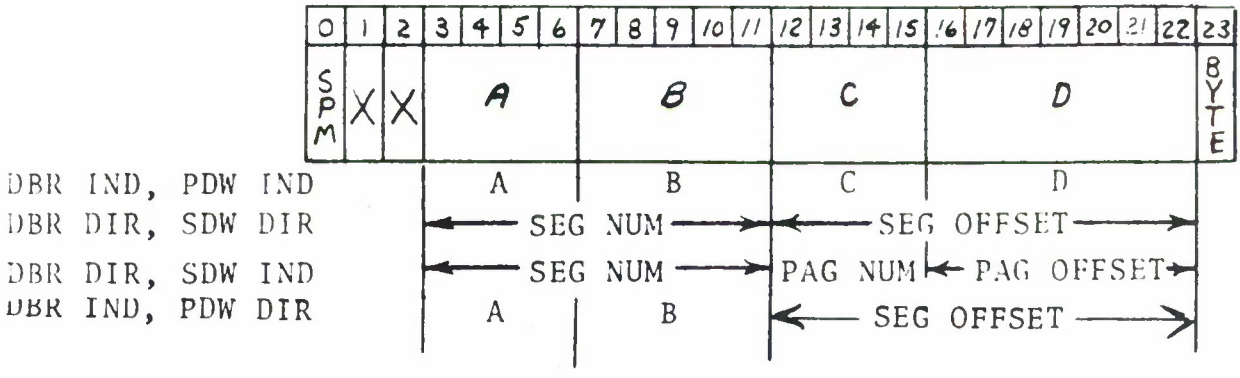


Figure 5A. VIRTUAL MEMORY ADDRESS

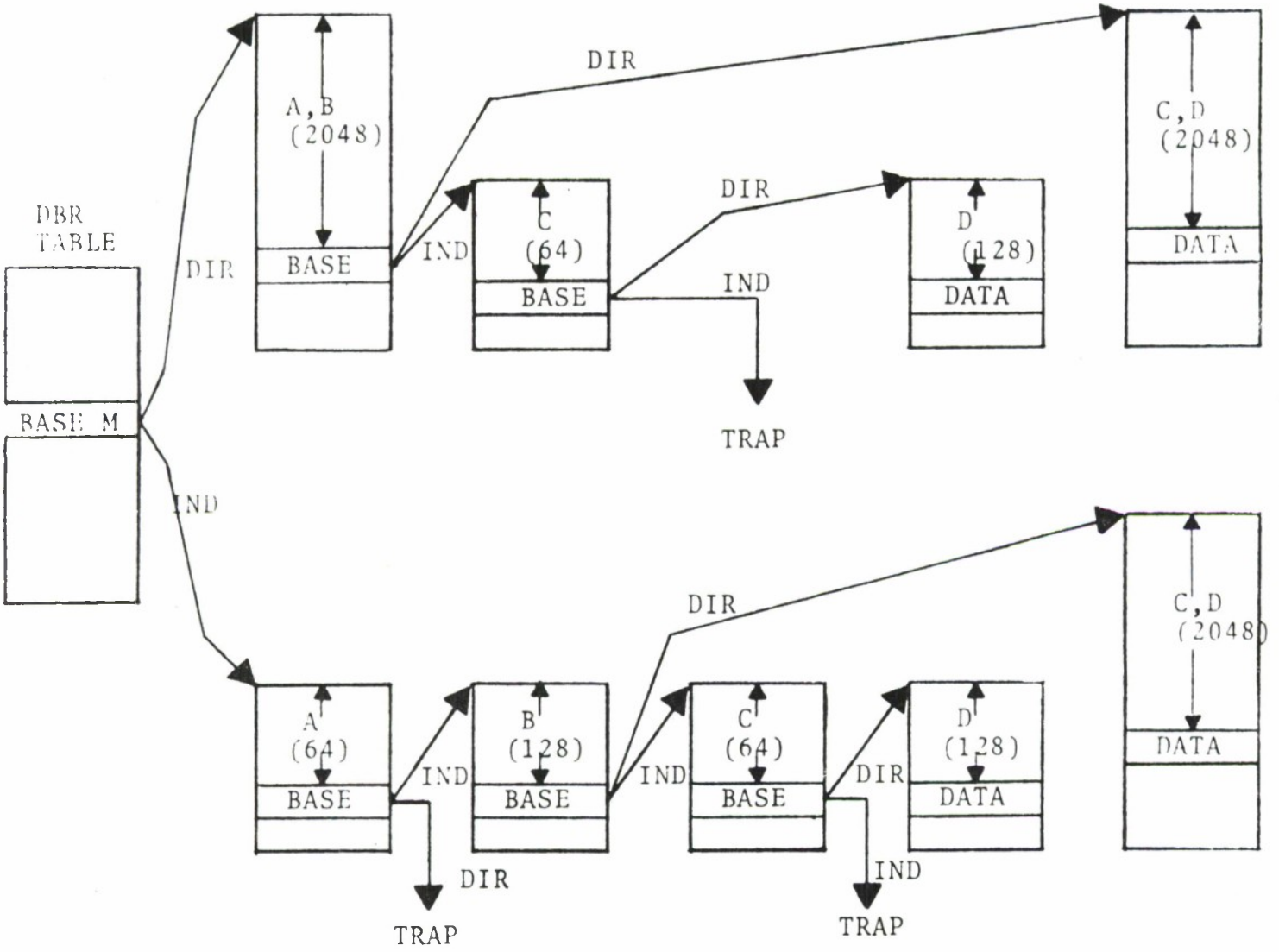


Figure 5B. VIEW OF MEMORY

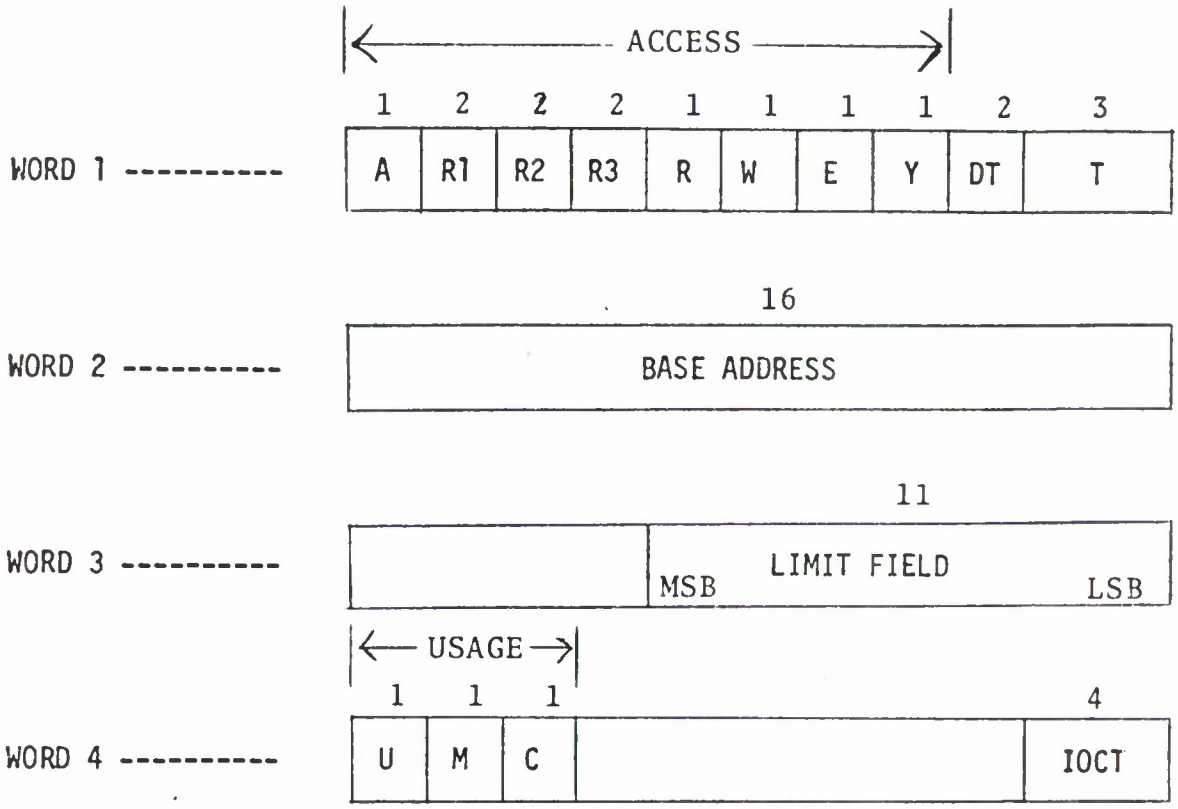


Figure 6. Memory Descriptor Format

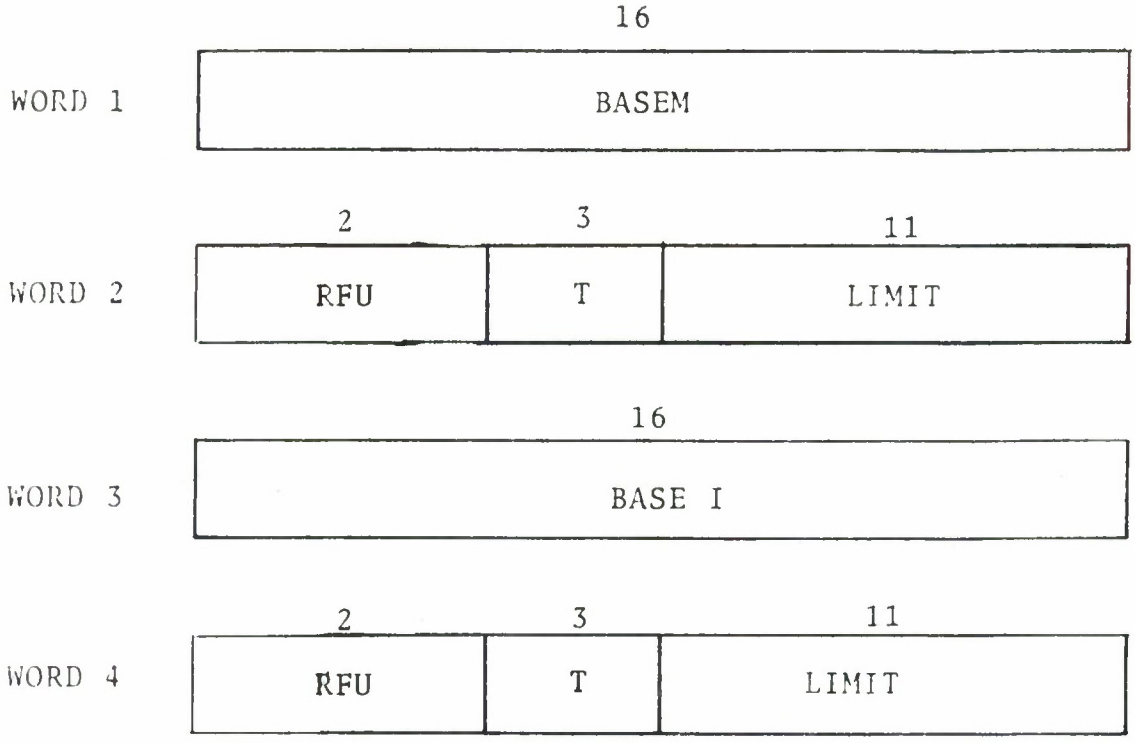


Figure 7. DBR FORMAT

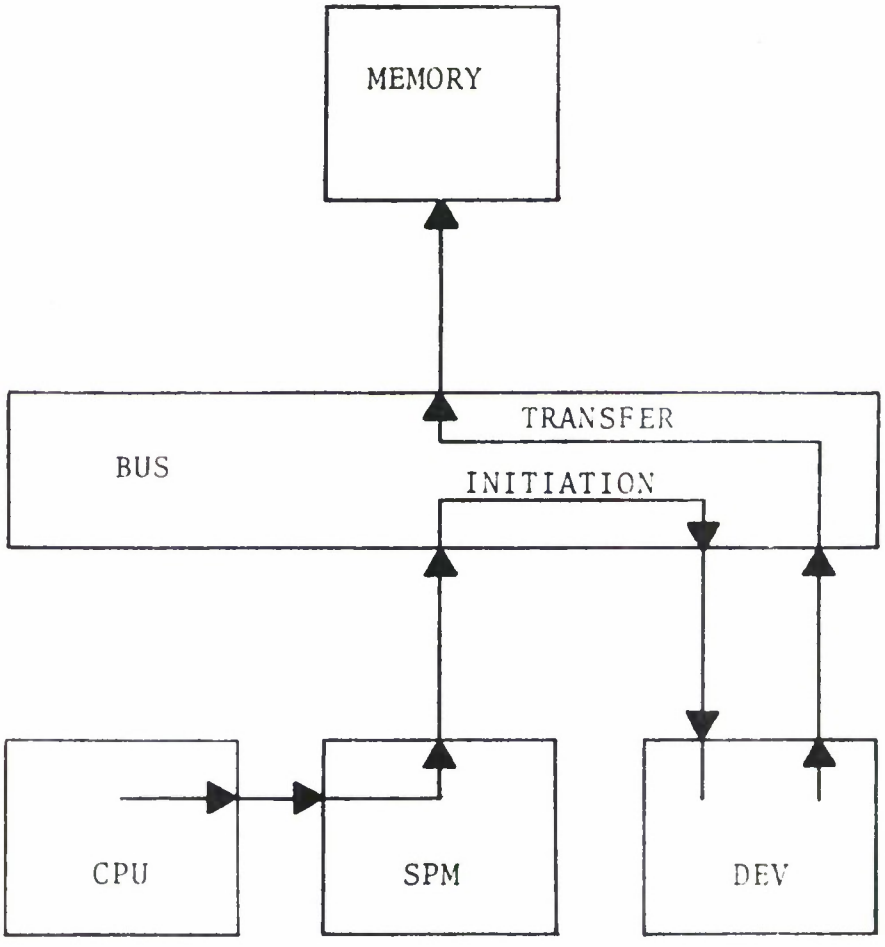


Figure 8. PREMAPPED I/O FLOW

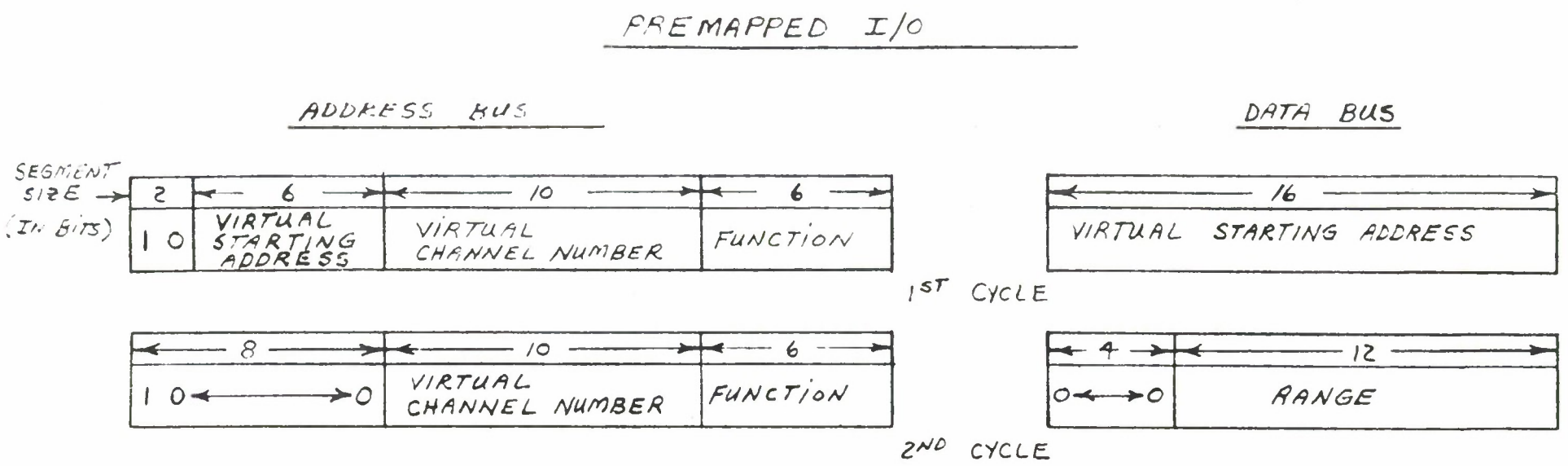


FIGURE 9 VIRTUAL ADDRESS (PREMAPPED I/O)

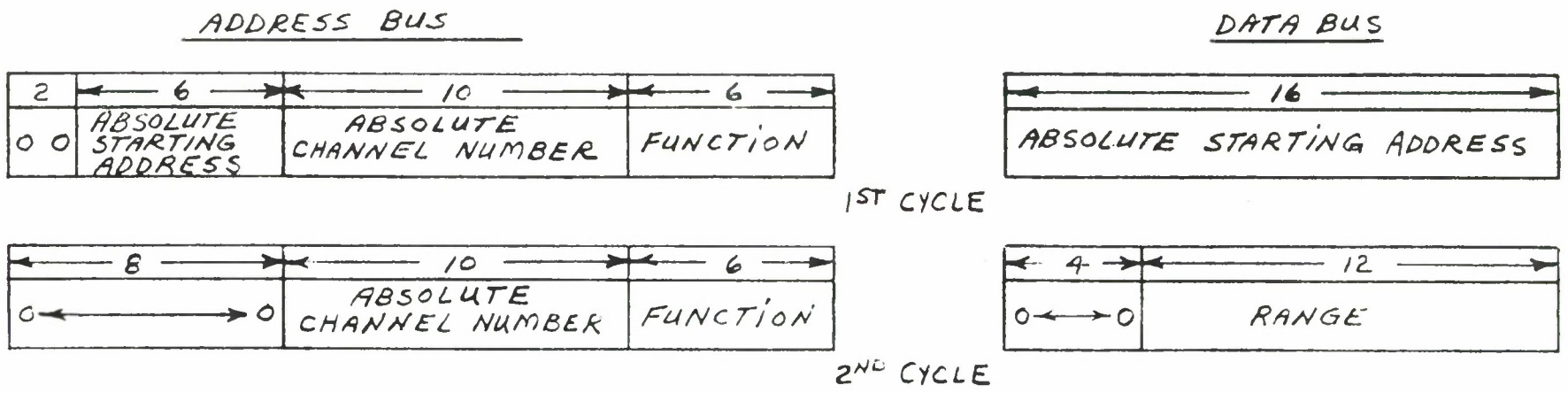


FIGURE 10. ABSOLUTE ADDRESS (PREMAPPED I/O)

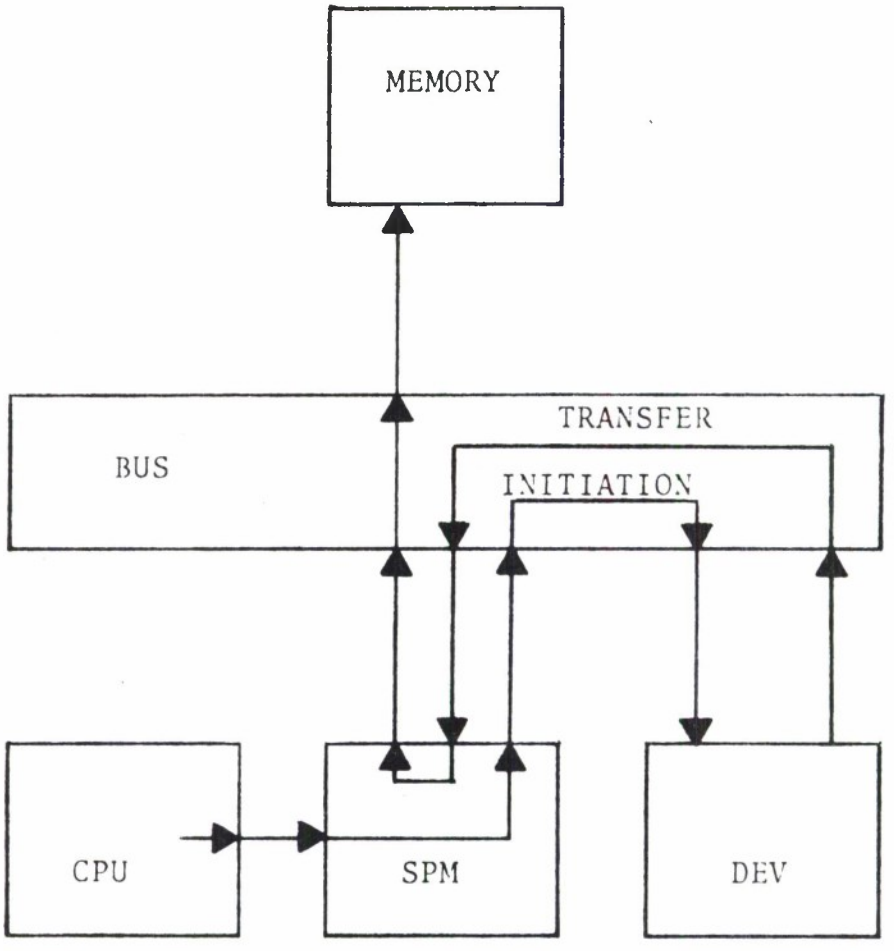


Figure 11. MAPPED I/O FLOW

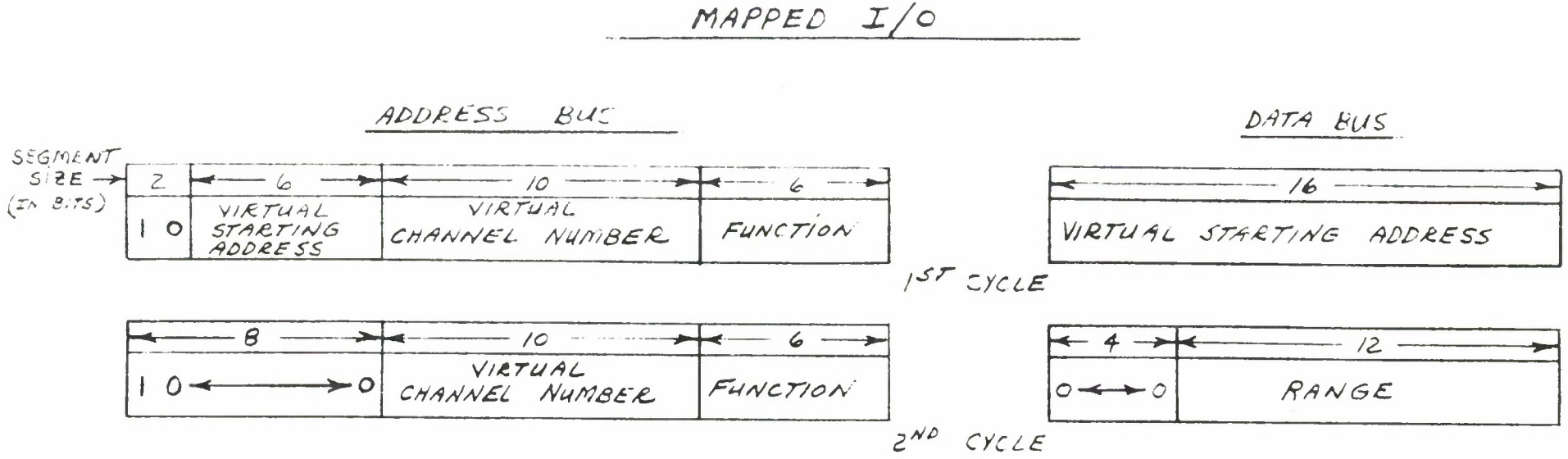


FIGURE 12 VIRTUAL ADDRESS (MAPPED I/O)

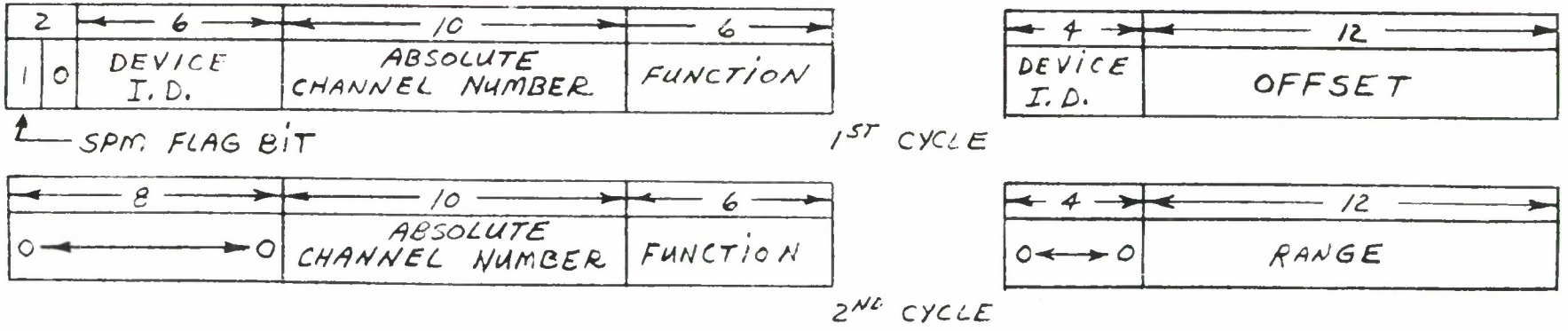


FIGURE 13. ABSOLUTE ADDRESS (MAPPED I/O)

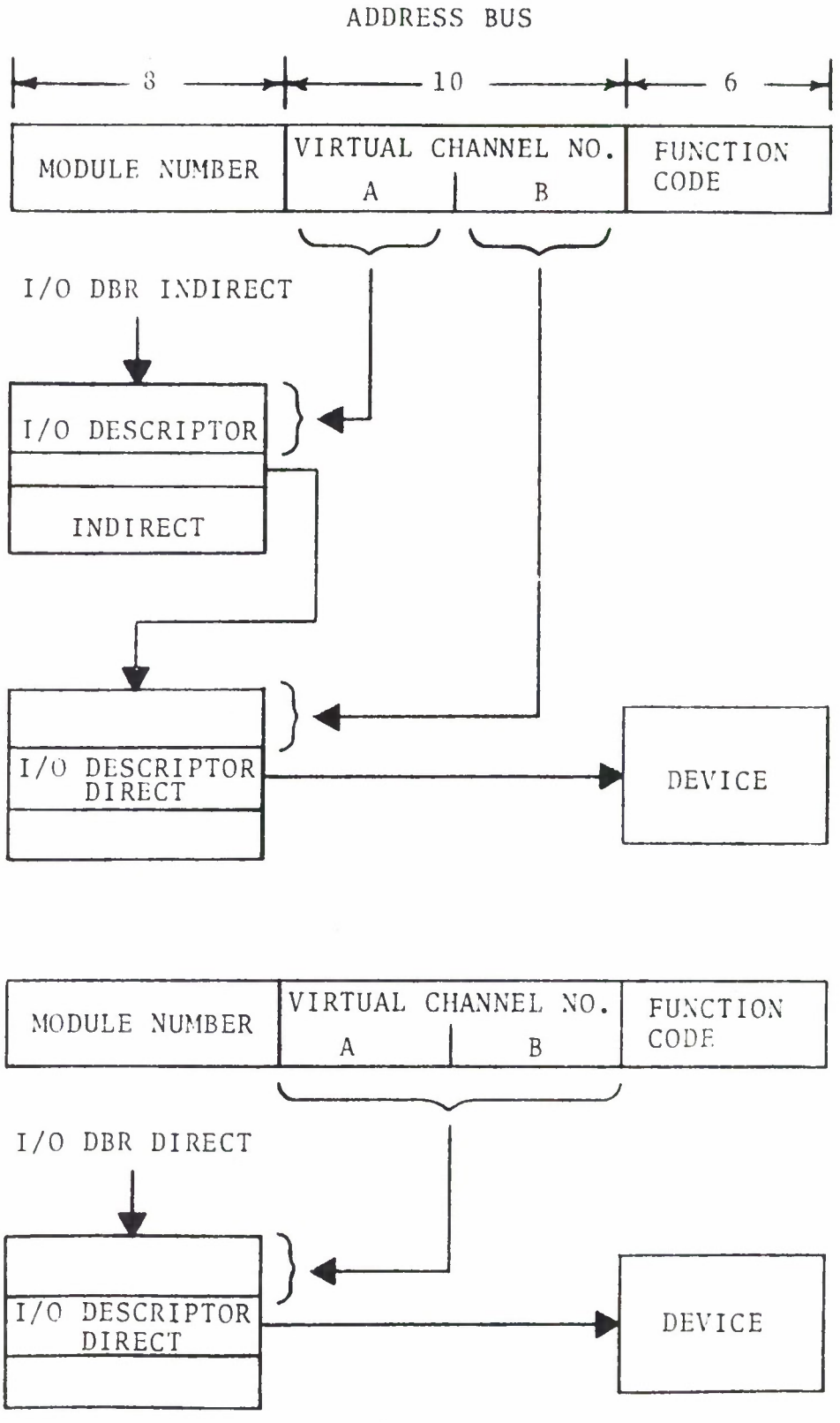


Figure 14. VIRTUAL ADDRESS TRANSLATION

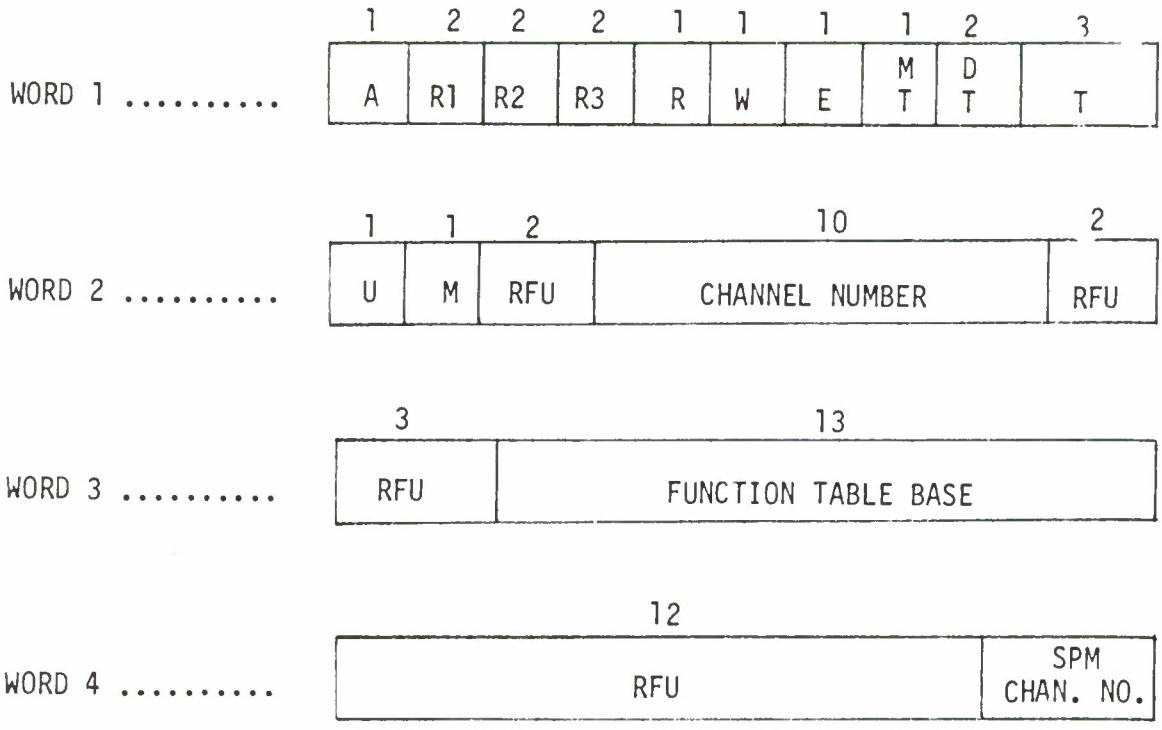


FIGURE 15. DIRECT I/O DESCRIPTOR FORMAT

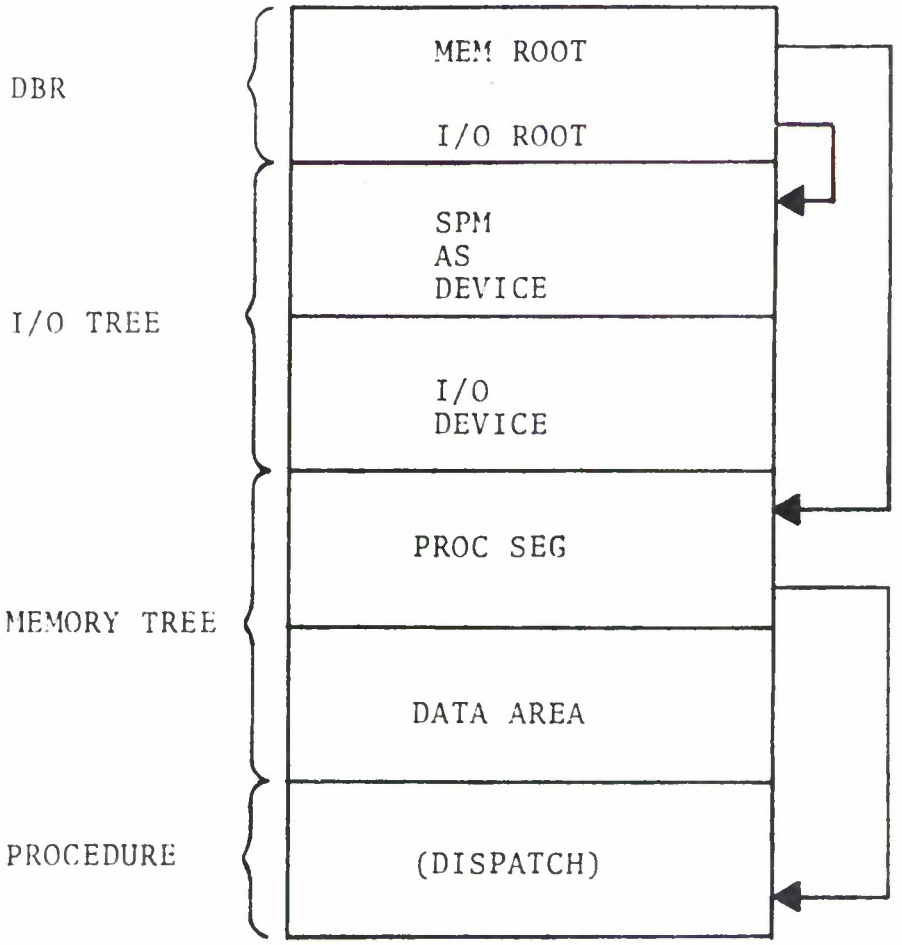


Figure 16. BOOTSTRAP

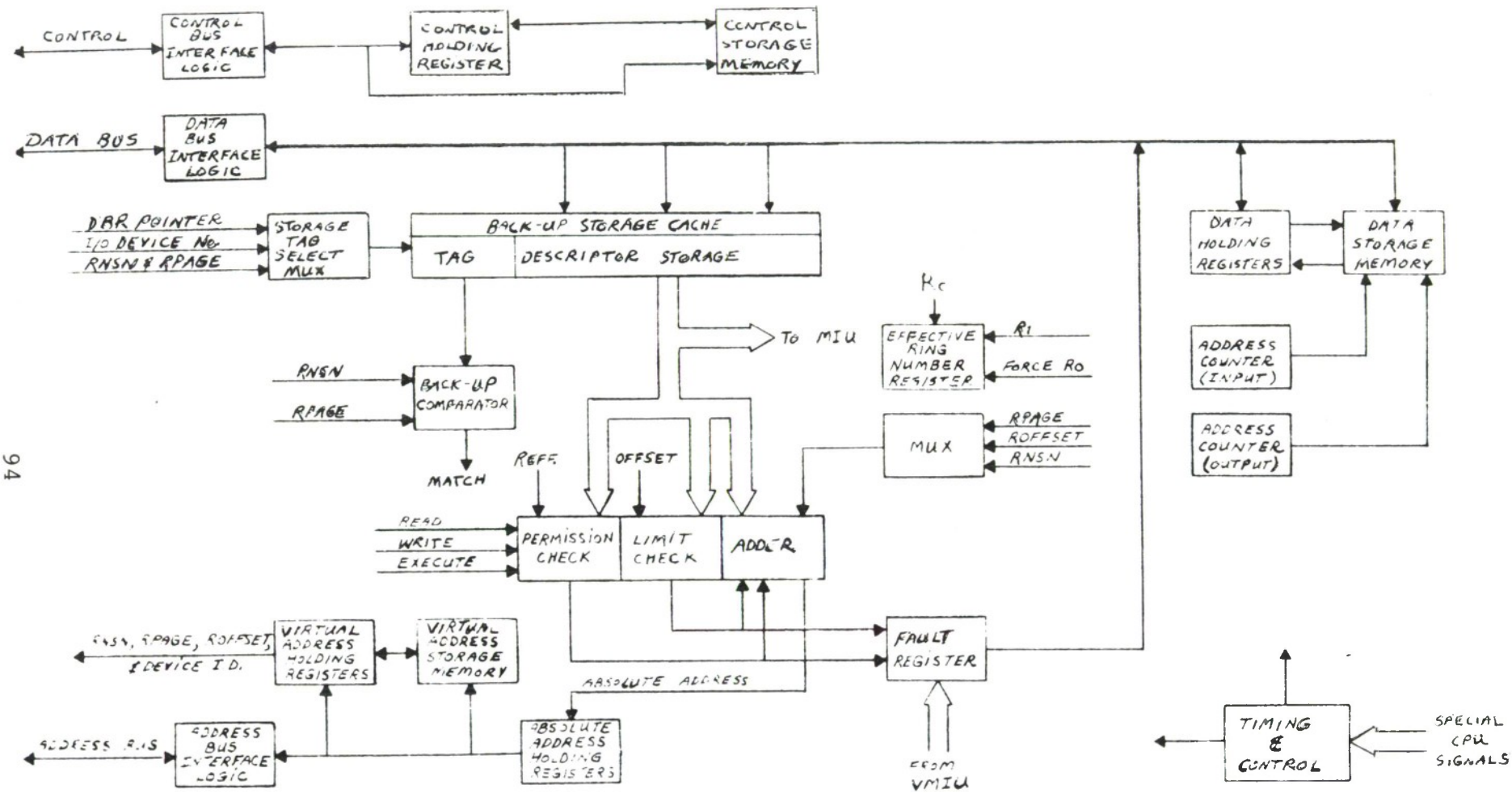


FIGURE 17. SPM BLOCK DIAGRAM

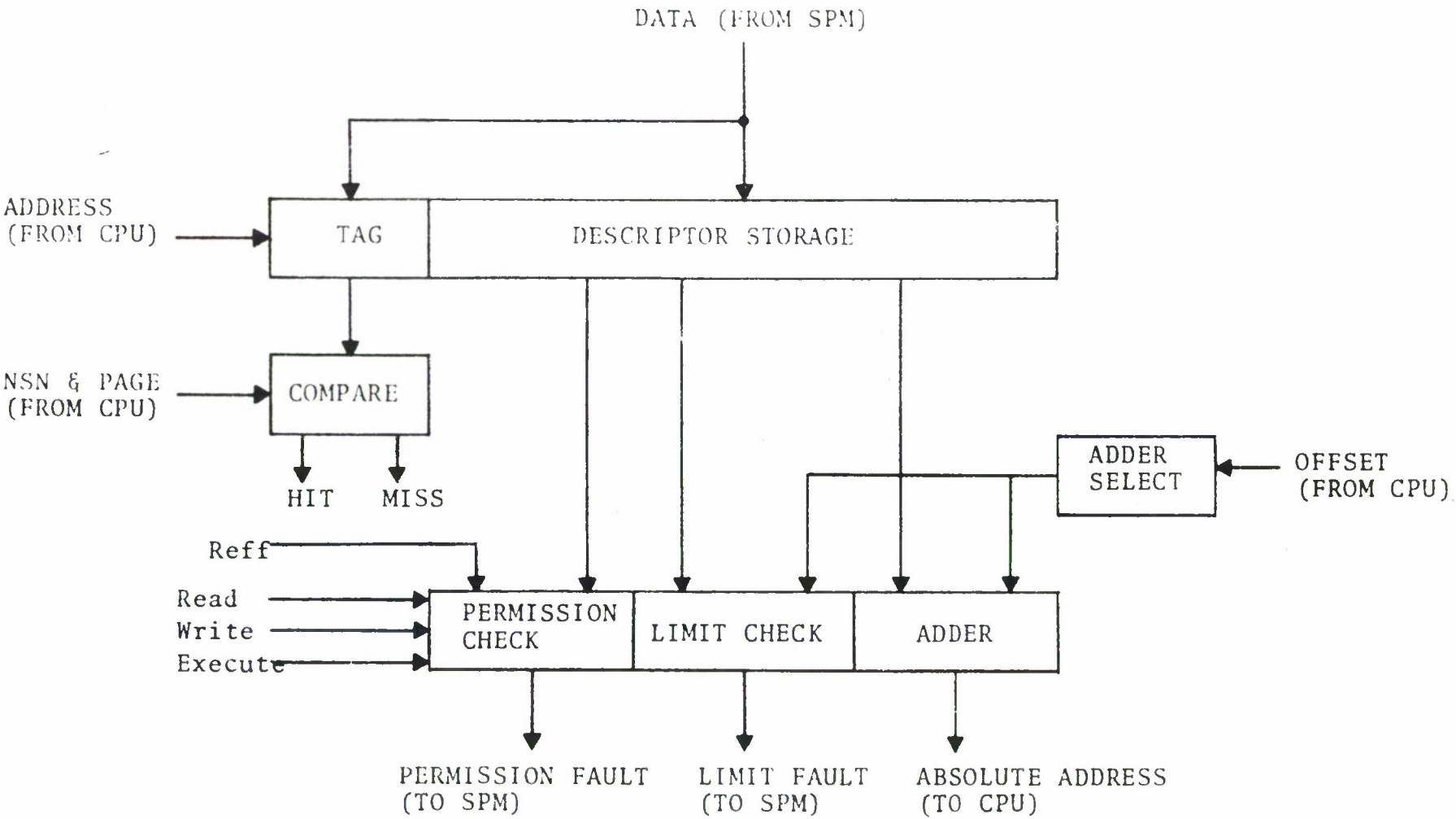


Figure 18. VMIU BLOCK DIAGRAM